

Universidad Carlos III de Madrid  
Escuela Politécnica Superior  
Departamento de Ingeniería Informática



TRABAJO DE FIN DE GRADO:

**MONITORIZACIÓN VISUAL DE UN  
CONDUCTOR MEDIANTE UNA CÁMARA  
ESTÉREO**

---

Autor: Blanco Rodríguez, Alejandro

Tutores: Sanchis de Miguel, Araceli  
Ledezma Espino, Agapito

Septiembre 2016

# Agradecimientos

Este proyecto pone fin a cuatro años de carrera que han pasado rápido pero intensamente. Me gustaría destacar y agradecer a las personas junto a las que todos estos años he vivido tanto buenos como malos momentos, pero todos merecedores de ser recordados.

En primer lugar, mi madre Ángeles y mi hermana Beatriz, que me han soportado durante toda mi vida y especialmente durante la carrera. Les agradezco que me hayan ayudado en todo lo que estuviera en su mano.

Agradezco también enormemente a mis tutores Araceli y Agapito, quienes siempre han estado disponibles cuando necesitaba ayuda con el proyecto y además he tenido el placer de contar con ellos como profesores durante mi aprendizaje en la universidad. Junto con ellos me gustaría agradecer a todas las integrantes del departamento CAOS por hacerme sentir como uno más durante los últimos meses, especialmente a mis compañeros de laboratorio.

Para finalizar, a todas las personas que han conseguido hacer mi estancia en Madrid única durante estos cuatro años, en especial a Victor, Saúl y Juanjo. Con ellos he pasado la mayoría de mi tiempo de clases y prácticas y parece ser que todavía me soportan. También a mis otros compañeros de carrera Juanlu, Alberto, Álvaro, Rubén, Sandra y Alejandro, con los que no han faltado buenos momentos. Por último a mi compañera no sólo de estudios, sino también de piso, Sandra; junto con la que he sobrevivido a mi etapa universitaria.

Gracias a todos.

# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación y objetivos . . . . .	2
1.2. Descripción del problema . . . . .	2
1.3. Alcance del proyecto . . . . .	5
1.4. Restricciones . . . . .	6
1.5. Marco regulador . . . . .	6
1.6. Entorno operacional . . . . .	7
1.7. Organización de la memoria . . . . .	8
<b>2. Estado del arte</b>	<b>9</b>
2.1. Sistemas Asistencia a la Conducción . . . . .	9
2.1.1. Sistemas de asistencia con control sobre el vehículo . . . . .	10
2.1.2. Sistemas de asistencia informativos . . . . .	14
2.2. Modelado de Emociones en el usuario . . . . .	17
2.2.1. Modelo Plutchik . . . . .	17
2.2.2. Modelo de Russell . . . . .	18
2.2.3. Modelo de Lang . . . . .	19
2.3. Detección de las emociones (ZenCar) . . . . .	20
2.3.1. Respuesta cardíaca . . . . .	20
2.3.2. Ondas cerebrales . . . . .	21
2.3.3. Parámetros del vehículo . . . . .	23
<b>3. Análisis del sistema</b>	<b>25</b>
3.1. Especificación de requisitos . . . . .	25
3.1.1. Requisitos funcionales . . . . .	26
3.1.2. Requisitos no funcionales . . . . .	30
3.2. Casos de uso . . . . .	32
3.2.1. Descripción tabular de casos de uso . . . . .	32
<b>4. Arquitectura y diseño del sistema</b>	<b>36</b>
4.0.1. Módulo interfaz gráfica . . . . .	37
4.0.2. Módulo gestión de <i>frames</i> . . . . .	37
4.0.3. Módulo seguimiento ocular . . . . .	37
4.0.4. Módulo Alarmas . . . . .	38
4.0.5. Módulo sockets . . . . .	38
<b>5. Implementación del sistema</b>	<b>39</b>
5.1. Funcionamiento del sistema . . . . .	40

5.1.1.	Flujo de imágenes . . . . .	41
5.1.2.	Detección de esqueletos . . . . .	43
5.1.3.	Detección facial . . . . .	44
5.1.4.	Monitorización del usuario . . . . .	44
5.1.5.	Comunicación mediante sockets . . . . .	56
<b>6.</b>	<b>Resultados y evaluación</b>	<b>58</b>
6.1.	Reconocimiento facial . . . . .	59
6.2.	Reconocimiento Ocular . . . . .	60
6.3.	Alarmas . . . . .	62
6.4.	Pruebas en el simulador . . . . .	64
<b>7.</b>	<b>Planificación del trabajo y entorno socioeconómico</b>	<b>66</b>
7.1.	Metodología de desarrollo Software . . . . .	66
7.2.	Planificación . . . . .	67
7.3.	Presupuesto . . . . .	70
7.3.1.	Costes Parciales . . . . .	70
7.3.2.	Coste Total . . . . .	71
7.4.	Impacto del proyecto . . . . .	72
<b>8.</b>	<b>Conclusiones y trabajos futuros</b>	<b>73</b>
8.1.	Conclusiones técnicas . . . . .	73
8.2.	Conclusiones personales . . . . .	74
8.3.	Trabajos futuros . . . . .	74

## Bibliografía



# Índice de figuras

1.	Muertes por accidentes en el mundo en los ultimos años (Fuente: <a href="http://www.who.int">www.who.int</a> ).	I
2.	Plataforma de simulación basada en el simulador <i>STISIM Drive</i> .	II
3.	Cámara <i>Kinect</i> v2.0.	III
4.	Diagrama del proyecto global donde se utilizará el sistema de monitorización.	IV
5.	Diagrama de módulos de la arquitectura del sistema.	V
6.	Diagrama de flujo de detección de la cara del usuario.	V
7.	Imagen captada por la cámara infrarroja	VI
8.	Ejemplo de la cara del conductor detectada por la cámara infrarroja.	VI
9.	Imagen captada por la cámara de alta definición.	VII
10.	Puntos de la cara detectados por la cámara de alta definición.	VIII
11.	Proceso al que se someten los <i>frames</i> para detectar la posición de los ojos.	VIII
12.	Imágenes de los sujetos de pruebas.	IX
13.	Deaths occasioned by traffic accidents in the world in the recent years (Source: <a href="http://www.who.int">www.who.int</a> ).	I
14.	Simulation platform based on the <i>STISIM Drive</i> simulator.	II
15.	<i>Kinect</i> v2.0 camera.	III
16.	Diagram of the global project where the monitoring system will be used.	IV
17.	Module diagram of the system's architecture.	V
18.	Detection of the user's face process.	V
19.	Image captured by the infrared camera.	VI
20.	Example of the user's face detected by the infrared camera.	VI
21.	Image captured by the high-definition camera.	VII
22.	Face points detected by the high-resolution camera.	VIII
23.	Process to detect the position of the eyes.	VIII
24.	Images of the test subjects.	IX
1.1.	Muertes por accidentes en todo el mundo en los ultimos años (Fuente: <a href="http://www.who.int">www.who.int</a> ).	1
1.2.	Plataforma de simulación basada en el simulador <i>STISIM Drive</i> .	3
1.3.	Cámara <i>Kinect</i> v2.0.	3
1.4.	Diagrama del proyecto global donde se utilizará el sistema de monitorización.	4
2.1.	Control de crucero adaptativo (Fuente: <a href="http://www.motor.mapfre.es">www.motor.mapfre.es</a> ).	10
2.2.	Sistema de frenado de emergencia (Fuente: <a href="http://www.telegraph.co.uk">www.telegraph.co.uk</a> ).	11
2.3.	Asistente de mantenimiento en el carril (Fuente: <a href="http://www.motor.mapfre.es">www.motor.mapfre.es</a> ).	11
2.4.	Asistente para atascos de tráfico (Fuente: <a href="http://www.xataka.com">www.xataka.com</a> ).	12
2.5.	Sistema de aparcamiento automático (Fuente: <a href="http://www.diariomotor.com">www.diariomotor.com</a> ).	12
2.6.	Luces dinámicas en curva (Fuente: <a href="http://www.car-lighting-selector.com">www.car-lighting-selector.com</a> ).	13
2.7.	Asistente de luces en carretera (Fuente: <a href="http://www.circulaseguro.com">www.circulaseguro.com</a> ).	13

2.8.	Detección de ángulo muerto (Fuente: <a href="http://www.autopista.es">www.autopista.es</a> ).	14
2.9.	Asistencia de visión nocturna (Fuente: <a href="http://www.circulaseguro.com">www.circulaseguro.com</a> ).	15
2.10.	Reconocimiento de señales (Fuente: <a href="http://www.circulaseguro.com">www.circulaseguro.com</a> ).	15
2.11.	Detección de fatiga (Fuente: <a href="http://www.telematicsnews.info">www.telematicsnews.info</a> ).	16
2.12.	Detección de fatiga (Fuente: <a href="http://www.motorpasionfuturo.com">www.motorpasionfuturo.com</a> ).	16
2.13.	Modelo de Plutchik (Fuente: <a href="http://www.vivirmejorconmigo.com">www.vivirmejorconmigo.com</a> ).	17
2.14.	Modelo de Russell (Fuente: <a href="http://www.redalyc.org">www.redalyc.org</a> ).	18
2.15.	Lang (Fuente: <a href="http://www.catarina.udlap.mx">www.catarina.udlap.mx</a> ).	19
2.16.	Estudio cardíaco en conductores españoles (Fuente: <a href="http://www.unespa.es">www.unespa.es</a> ).	21
2.17.	Estudio de la activación en conductores españoles (Fuente: <a href="http://www.unespa.es">www.unespa.es</a> ).	22
2.18.	Estudio de la atención en conductores españoles (Fuente: <a href="http://www.unespa.es">www.unespa.es</a> ).	22
2.19.	Estudio indicador Zen en conductores españoles (Fuente: <a href="http://www.unespa.es">www.unespa.es</a> ).	23
2.20.	Estudio combinado en conductores españoles (Fuente: <a href="http://www.unespa.es">www.unespa.es</a> ).	24
4.1.	Diagrama de módulos de la arquitectura del sistema.	36
5.1.	Diagrama de flujo de detección de la cara del usuario.	40
5.2.	Imagen tomada por la cámara infrarroja de la posición a ocupar por el conductor en el simulador.	41
5.3.	Imagen tomada por la cámara de alta definición de la posición a ocupar por el conductor en el simulador.	42
5.4.	Botones de selección de modo de imagen (señalados en un círculo rojo).	42
5.5.	Cámara Kinect v1 (Fuente: <a href="http://www.users.dickinson.edu">www.users.dickinson.edu</a> ).	43
5.6.	Base de datos de esqueletos utilizada por <i>Kinect</i> para detectar al usuario.	43
5.7.	Imagen del conductor captada por la cámara infrarroja.	45
5.8.	Haz de luz infrarroja que proyecta la cámara.	45
5.9.	Ejemplo de la cara del conductor detectada por la cámara infrarroja.	46
5.10.	Posiciones de la cabeza (Fuente: <a href="http://65.media.tumblr.com">65.media.tumblr.com</a> ).	47
5.11.	Imagen del conductor captada por la cámara de alta definición.	48
5.12.	Puntos de la cara detectados por la cámara de alta definición.	49
5.13.	Región de interés en una imagen captado por la cámara de alta definición.	50
5.14.	Ojo obstaculizado por la posición de la cabeza (señalado por círculo rojo).	50
5.15.	Puntos de la cara detectados por la cámara infrarroja y de alta definición.	51
5.16.	Región de interés en escala de grises.	52
5.17.	Región de interés tras someterla a un filtro de blanco y negro.	52
5.18.	Región de interés tras someterla a un filtro de erosión.	53
5.19.	Región de interés con centro de masas detectado.	53
5.20.	Proceso al que se someten los <i>frames</i> para detectar la posición de los ojos.	54
5.21.	Eventos registrados de forma intercalada [33].	55
5.22.	Eventos registrados consecutivamente [33].	55
5.23.	Estructura global del sistema.	57
6.1.	Imágenes de los sujetos de pruebas.	58
6.2.	Cámara <i>Kinect</i> colocada en el simulador (señalada con círculo rojo).	64
6.3.	Mapa de profundidad del entorno del simulador.	65
7.1.	Diagrama de la metodología en cascada.	67
7.2.	Planificación Gantt del proyecto.	69

# Resumen

La seguridad en las carreteras ha sido siempre una tema de gran importancia en la sociedad y en los últimos años se han observado importantes avances tecnológicos para garantizarla. Los accidentes de tráfico son la mayor causa de muerte entre las personas de entre 15 y 29 años y en la actualidad se ha llegado a la cifra de 1,25 millones de muertes por año a nivel mundial debido a colisiones de tráfico [1]. El número de muertes ha comenzado a estabilizarse desde 2007 a causa de los avances que han permitido mejorar la seguridad vial (Figura 1.1), es decir, los Sistemas Avanzados de Asistencia a la Conducción o ADAS (*Advanced Driver Assistance Systems*).

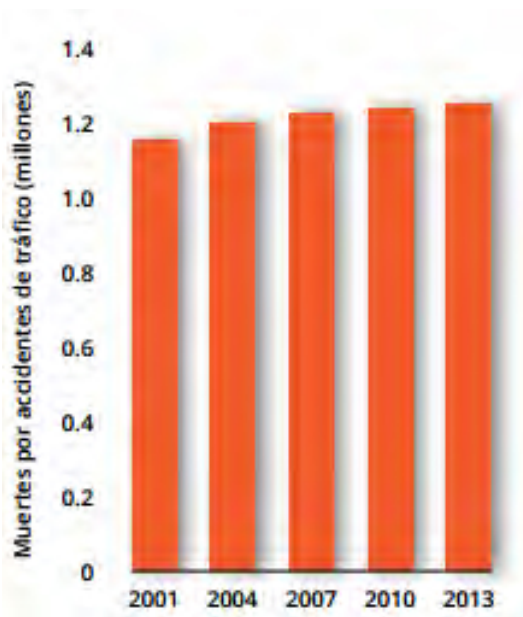


Figura 1: Muertes por accidentes en el mundo en los ultimos años (Fuente: [www.who.int](http://www.who.int)).

Por este motivo, hay que tener en cuenta la importancia de estos sistemas, que no sólo ayudan a las personas dentro del vehículo, si no que aumentan la seguridad general en la carretera, ya que la mitad de las víctimas de los accidentes mortales son peatones, ciclistas o motoristas, los cuales no cuentan con sistemas de seguridad tan sofisticados como los que puede incorporar un coche.

Gran parte de los accidentes de tráfico son ocasionados por distracciones del conductor o por falta de atención, las cuales aumentan con el grado de fatiga o el no prestar atención únicamente a la conducción. La mayoría de sistemas de ayuda a la conducción pueden evitar accidentes en estas situaciones de falta de atención, pero, dado que del 20 % al 30 %

de los accidentes de tráfico se relacionan con el cansancio [2], se hace necesaria la existencia de sistemas que monitoricen el comportamiento del conductor y eviten o adviertan de este estado de fatiga. De esta manera no se deja la seguridad del coche únicamente en manos de estos sistemas que evitan los accidentes en el momento en el que van a ocurrir. Es necesario, además, detectar y avisar al conductor de las situaciones de peligro causadas por la fatiga. Lo más común es que, el conductor, aunque es consciente de su cansancio, infravalora el riesgo de quedarse dormido al volante, pudiendo causar un accidente de gran gravedad en caso de cerrar los ojos durante unos segundos [3].

La fatiga no es el único factor a tener en cuenta, como se ha dicho antes, también pueden existir distracciones por llevar a cabo otras actividades mientras se conduce o por falta de atención debido a las múltiples emociones que puede experimentar un ser humano. Nuevamente, el conductor es el punto central al que hay que prestar atención para determinar en qué estado se encuentra y poder evitar situaciones peligrosas.

En este trabajo fin de grado se realizará una primera aproximación a un sistema de monitorización del usuario, el cual podrá ser incorporado más adelante a un sistema de asistencia a la conducción completo. La necesidad de un sistema de reconocimiento facial comienza con un Sistema Avanzado de Asistencia a la Conducción utilizado por el grupo de investigación *CAOS* (Control, Aprendizaje y Optimización de Sistemas) de la Universidad Carlos III de Madrid [4], para el cual se está desarrollando un sistema basado en agentes con un alto nivel de razonamiento. Este sistema se utiliza para realizar simulaciones de conducción en un entorno urbano poniendo especial atención en la seguridad vial. El simulador utilizado es *STISIM Drive* y se puede ver en la Figura 1.2 [6].



Figura 2: Plataforma de simulación basada en el simulador *STISIM Drive*.

En este sistema multiagente, como su propio nombre indica, existen varios agentes con diferentes roles, los cuales se pueden separar en dos grupos. Por un lado, los que tienen como objetivo la recogida de información y por otro, los utilizados para el análisis y fusión de esta información para definir qué situaciones dentro de la simulación pueden resultar peligrosas. Dentro del primer grupo se puede recoger información del entorno de simulación, como peatones o señales de tráfico, información del vehículo, como la velocidad o el grado de presión en los pedales, o información acerca del conductor. Es en este último subgrupo donde se requiere el uso de un sistema de reconocimiento facial para recoger esta información que pueda utilizar el sistema multiagente para conseguir definir situaciones más complejas y por lo tanto, más precisas.



Figura 3: Cámara *Kinect* v2.0.

Para llevar a cabo esta tarea se utilizará el dispositivo *Kinect* v2.0 (Figura 1.3) y se implementará una aplicación utilizando las herramientas proporcionadas por el fabricante del mismo. El sistema desarrollado en este proyecto se limitará a monitorizar al conductor y enviar la información al equipo donde se ejecuta el simulador de conducción. La información básica es detectada por el dispositivo *Kinect* y, como añadido, se implementa un método de detección ocular no incluido entre las funcionalidades del dispositivo. Además se utilizará un sistema de alarmas implementado anteriormente en un trabajo fin de máster [33] añadiendo nuevas alarmas propias, que aportarán más información al sistema multiagente. Este sistema se dedica a la fusión y análisis de los datos, tras lo que determina la peligrosidad de la situación de tráfico y toma la decisión de si activar o no una alarma.

En la figura 1.4 se pueden observar los componentes del proyecto global dentro del que funcionará el sistema desarrollado en este trabajo fin de grado.

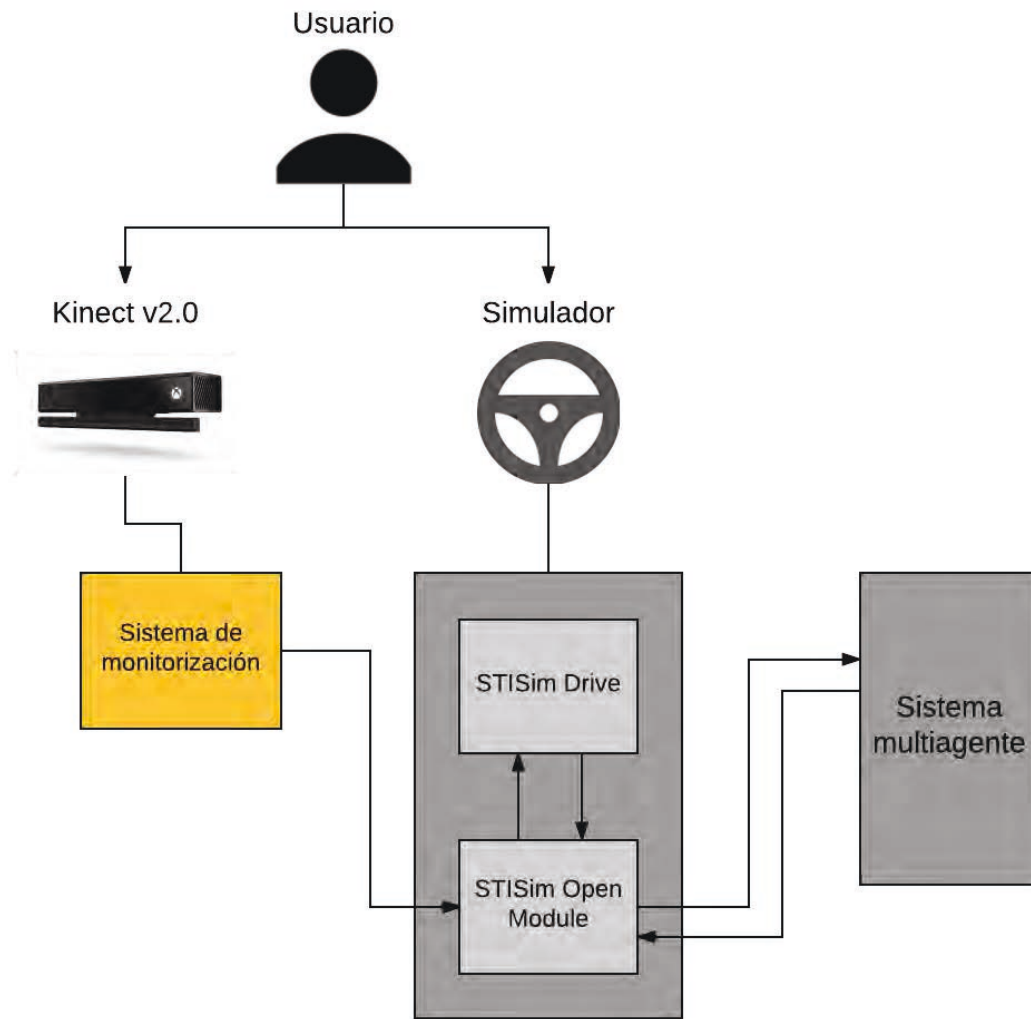


Figura 4: Diagrama del proyecto global donde se utilizará el sistema de monitorización.

Por lo tanto, sintetizando lo anterior, el objetivo de este trabajo fin de grado se divide en los siguientes:

- Extracción de la información obtenida del reconocimiento facial llevado a cabo por el dispositivo Kinect v2.0.
- Implementación de un método para detectar la posición de las pupilas.
- Generar información auxiliar mediante un sistema de alarmas.
- Comunicación de la información en tiempo real entre equipos.

En el sistema se pueden encontrar cinco módulos principales que interactúan entre sí: la interfaz gráfica, la gestión de *frames*, el seguimiento ocular, el sistema de alarmas y la conexión mediante sockets. Cada uno de estos componentes representa una de las funcionalidades principales del sistema. En la figura 4.1 se puede observar cómo están interconectados estos sistemas entre sí.

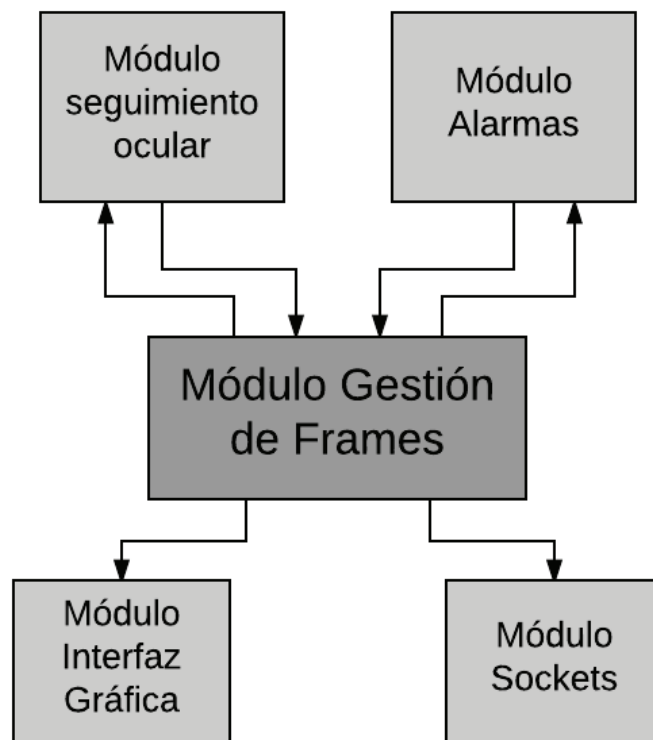


Figura 5: Diagrama de módulos de la arquitectura del sistema.

Se ha utilizado como modelo el proyecto creado por el departamento de Automática para el trabajo fin de máster *Detección del estado emocional de un conductor mediante el uso de imágenes infrarrojas* [33]. El funcionamiento del programa se divide en varias etapas que se presentan a continuación en la figura 5.1.

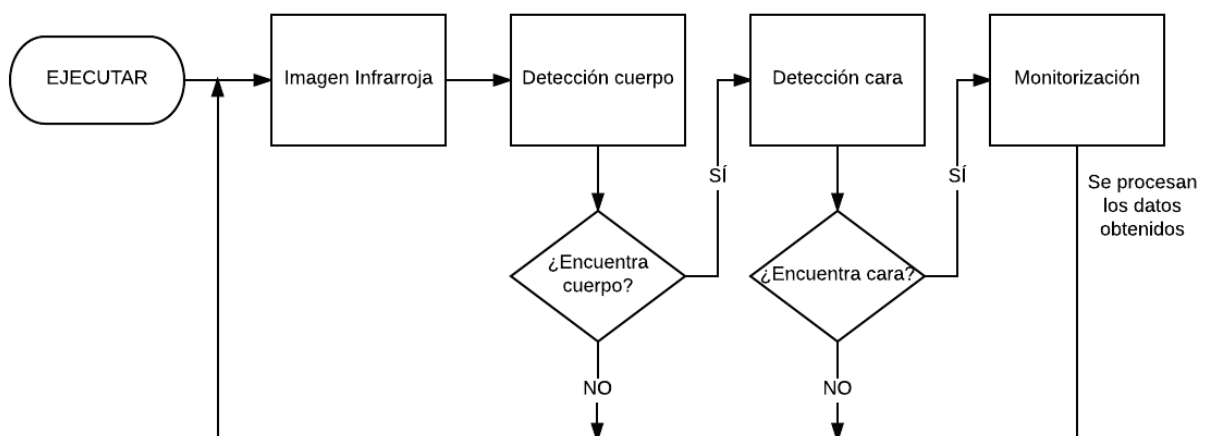


Figura 6: Diagrama de flujo de detección de la cara del usuario.

Aunque la simulación de conducción se realiza en un entorno cerrado y con condiciones lumínicas controladas, hay que tener en cuenta las condiciones que se aplicarían en un entorno urbano real. Por lo tanto, se utilizará la cámara infrarroja que ofrece Kinect, de tal forma que la iluminación no suponga un problema a la hora realizar la monitorización. Esta cámara emite un haz infrarrojo y es capaz de detectar la reflexión de este haz sobre los cuerpos, de esta forma la radiación solar no interfiere en el proceso (figura 5.7).



Figura 7: Imagen captada por la cámara infrarroja

Utilizando la información captada por la cámara se puede mostrar en la interfaz gráfica los puntos de interés sobre la cara del usuario para conocer su posición en todo momento. Además se muestra también el estado de estos puntos de interés, tal y como se muestra en la figura 5.9.



Figura 8: Ejemplo de la cara del conductor detectada por la cámara infrarroja.



El siguiente aspecto que se va a monitorizar en el usuario es la posición de las pupilas. La cámara ofrece muchas características que suponen una cantidad de información considerable, pero se considera que conocer la posición de los ojos es esencial para garantizar la seguridad del conductor. Sin esta información el usuario puede tener la cabeza perfectamente alineada con la carretera y el sistema consideraría que no ocurre nada malo, mientras que el usuario puede estar mirando en otra dirección, creando una situación de peligro.

Para determinar la posición de la pupila se utilizan las imágenes captadas por la cámara a color de alta resolución (como la que se puede observar en la figura 5.11).



Figura 9: Imagen captada por la cámara de alta definición.

La cámara a color, al igual que la cámara infrarroja, reconoce una serie de puntos pertenecientes al rostro humano, pero, en este caso, al contar con una mayor resolución, se tiene acceso a muchos más puntos, los cuales se pueden observar en la figura 5.12 en color azul.



Figura 10: Puntos de la cara detectados por la cámara de alta definición.

Para detectar la posición de la pupila en una imagen captada por la cámara, se utilizan las herramientas proporcionadas por la librería *EmguCV* y se somete dicha imagen al proceso mostrado en la figura 5.20.

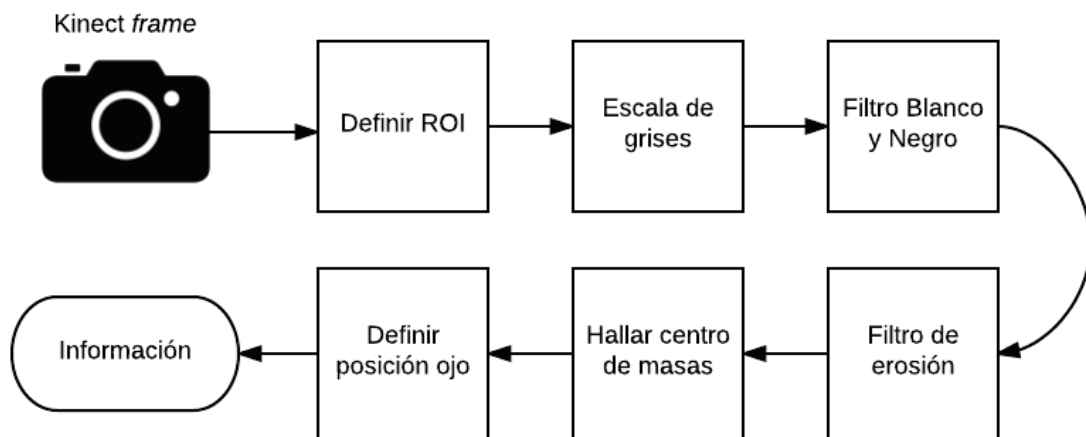


Figura 11: Proceso al que se someten los *frames* para detectar la posición de los ojos.

Para terminar la monitorización del usuario se propone utilizar la información recogida hasta ahora para crear nueva información, es decir, interpretar los datos recogidos de forma que pueda sacarse información nueva de ellos que puede ser de ayuda. Para ello se ha utilizado un sistema de alarmas. El sistema que se ha utilizado se ha desarrollado en el trabajo fin de máster *Detección del estado emocional de un conductor mediante el uso de imágenes infrarrojas* [33] y se le ha añadido un caso más teniendo en cuenta la posición de los ojos.

Para este sistema de alarmas se consideran los siguientes casos que pueden generar situaciones peligrosas:

- Pestañeos largos
- Boca abierta
- Giro de cabeza
- Habla
- Ingesta de bebidas
- Posición de los ojos

Como ya se ha explicado anteriormente, el sistema se conecta con el simulador de conducción mediante la implementación de un socket síncrono, utilizando una arquitectura cliente-servidor. El equipo en el que se encuentra esta aplicación actúa como servidor y activa un socket mientras espera por conexión. Una vez que el socket está abierto y el servidor espera para la conexión, se puede ejecutar el simulador en el equipo que actúa como cliente. En este momento el cliente envía un mensaje al servidor solicitando que éste envíe información sobre la monitorización del usuario. En el momento en que este mensaje es recibido, el servidor envía al cliente una línea de texto que contiene la información recopilada por el programa que se ha explicado en los puntos anteriores.

Para evaluar el funcionamiento de la aplicación se ha escogido a diez sujetos de pruebas de distinto género y distinta apariencia física, de esta forma se observa el comportamiento del sistema al monitorizar distintos sujetos con distintas características. En la figura 24 se pueden observar imágenes de los sujetos con los que se ha probado la efectividad del sistema.

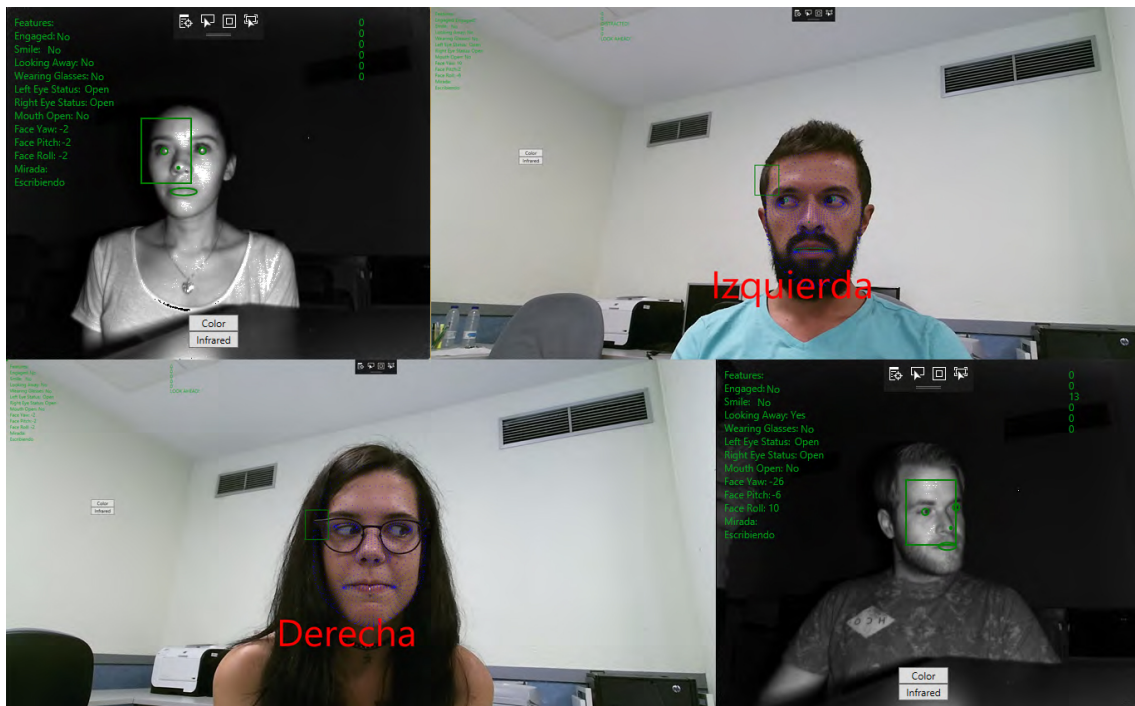


Figura 12: Imágenes de los sujetos de pruebas.

Para finalizar, se exponen las conclusiones a las que se ha llegado tras la implementación y desarrollo del trabajo fin de grado, así como las líneas de trabajo futuras hacia las que se puede dirigir este proyecto.

Una vez que se ha terminado el proyecto, se puede concluir que los objetivos principales sí han sido cumplidos. Se ha conseguido realizar la monitorización del usuario y controlar en todo momento la posición y situación de diferentes rasgos de la cara a tener en cuenta. Se ha encontrado una forma de recopilar esta información y mostrarla por pantalla. También se ha conseguido un método fiable para llevar a cabo el reconocimiento de estas características independientemente de las condiciones lumínicas mediante el uso de una cámara infrarroja. Además se ha generado información adicional basándose en los datos iniciales mediante el sistema de alarmas.

Como añadido se ha implementado el sistema de reconocimiento de la posición de las pupilas del usuario, característica que el dispositivo Kinect v2.0 no es capaz de detectar. Esta nueva funcionalidad puede ser utilizada en condiciones lumínicas favorables.

Otro de los objetivos de este proyecto era establecer la comunicación entre el sistema que controla la monitorización y el sistema donde se ejecuta la simulación de conducción. Este objetivo también se ha cumplido mediante la implementación de un socket síncrono que conecta los equipos que alojan estos sistemas y permite el intercambio de mensajes entre ellos.

Durante la realización de este trabajo se ha tomado conciencia de la magnitud y dificultad que lleva consigo un proyecto de estas dimensiones, teniendo que fijar los objetivos principales y planificando los tiempos de diseño, implementación, etc. También se ha tomado contacto con el trabajo llevado a cabo por el grupo de investigación CAOS de esta universidad, ayudando a desarrollar una pequeña parte de un proyecto mucho mayor. Durante el proceso se han adquirido ciertas habilidades de trabajo y aprendizaje necesarias para el futuro.

También vale la pena mencionar los conocimientos adquiridos acerca de la monitorización de usuarios y captura de movimiento mediante dispositivos como el que se ha usado durante el proyecto, aprendiendo a desenvolverse con las herramientas necesarias para su manipulación, así como los conocimientos sobre procesamiento de imágenes en un sistema que funciona en tiempo real.

Como conclusión técnica general, se han aplicado de forma satisfactoria los conocimientos adquiridos durante el grado a un problema relativo a la ingeniería informática.

En lo relativo a los trabajos futuros, el sistema desarrollado en este trabajo de fin de grado es una primera aproximación a la monitorización de un conductor de un vehículo y únicamente se controlan información bruta sobre el estado físico del usuario. En el futuro se puede profundizar en el ámbito de las emociones, las cuales, como ya se ha mencionado, tienen un impacto muy grande en la manera de comportarse de las personas y pueden influir perjudicialmente en la conducción. Mediante una detección facial más detallada se podrían llegar a tener en cuenta estas emociones en la conducción. Posiblemente se podría utilizar la información captada por la cámara de alta definición del dispositivo Kinect v2.0, aunque es posible que la expresiones faciales no sean lo suficientemente detalladas.

El paso a seguir a continuación de este trabajo sería la integración de la información captada por este sistema dentro del sistema multiagente desarrollado por el departamento CAOS. Esta información del conductor deberá ser utilizada dentro de la ontología del sistema multiagente y así éste podrá considerar situaciones mucho más complejas durante la simulación de conducción.

El otro aspecto a mejorar es el seguimiento de la posición de la pupilas, este se controla de forma independiente de la posición de la cabeza y sólo considera tres valores para la posición del ojo. Como trabajo futuro se propone encontrar otro método que permita conocer la posición de la pupila de manera precisa y la dirección de la mirada teniendo en cuenta la posición de la cabeza. Este método, además, para ser todavía más preciso deberá funcionar con las imágenes infrarrojas, para no depender de la iluminación y poder ser utilizado en cualquier situación mientras se conduce.

# Abstract

The security on the roads has been always a very important subject in the society and recently they have been noticed important technologic advances to garanty it. Traffic accidents are the most important cause of accidend between fifteen and twenty nine years old people and nowadays there have been 1,25 millions of deaths a year all over the world due to traffic crashed accidents [1]. The number of deaths have begun to get stabilished from 2007 because of advances which have allowed to improve road safety (Figure 1.1), that is to say, advanced driver assistance systems or ADAS (textitAdvanced Driver Assistance Systems).

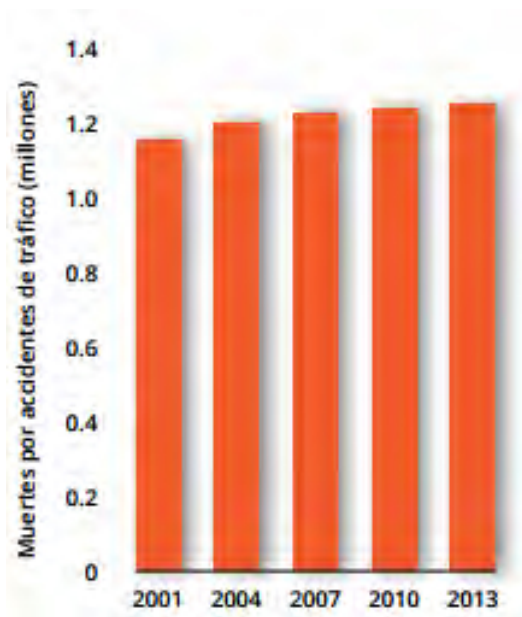


Figura 13: Deaths occasioned by traffic accidents in the world in the recent years (Source: [www.who.int](http://www.who.int)).

For this reason, it must be considered the importance of these systems, which not only help people into a vehicle, but also they increase the general road safety, because half of the fatal accident victims are pedestrians, cyclists or motorcyclist, who haven't safety systems so sophisticated than those ones which a car can incorporate.

Much part of traffic accidents are caused by distractions from the driver or by lack of attention, which is increased with the degree of fatigue or not paying attention only to the driving. Most of the support driving systems can avoid accidents in these cases of lack of attention, but, knowing that from the 20 % to the 30 % of traffic accidents are



related with the fatigue [2], it is necessary the existence of systems which monitor the behaviour and avoid or warn about this state of fatigue. In this way, safety is not left only in the hands of these systems that avoid accidents at the moment when they are going to happen. In addition, it is necessary detect and warn the driver about the dangerous situations caused by the fatigue. The most common is that, the driver, despite of being aware of his fatigue, underestimates the risk of falling asleep when he is driving and so he can cause an extremely serious accident in case of closing his eyes for some seconds.

Fatigue is not the only factor to consider, as it is said above, there can be also distractions because of doing other activities while driving or by lack of attention due to multiple emotions that a human being can experience. One more time, the driver is the central point to whom attention must be paid to determine his state and avoid dangerous situations.

In this end-of-degree project, it will be implemented a first approach to a user monitoring system, which will be able to be incorporated later to a more advanced assistance driver system. The need of a facial recognition begins with an Advanced Driver Assistance System used by the group of investigation *CAOS* (Control, Learning and System optimization) of Carlos III University in Madrid [4], for which it is being developed an agent-based system with a high level reasoning. This system is used to do driving simulations in an urban environment paying special attention in road safety. The simulator used is *STISIM Drive* and it can be seen in the figure 1.2 [6].



Figura 14: Simulation platform based on the *STISIM Drive* simulator.

In this multiagent system, as the name suggest, there are several agents with different roles that can be separated in two groups. On the one hand, those whose target is information gathering and on the other hand, those used to get information from simulation environment, such as pedestrians or traffic signs, vehicle information, as the speed or the pressure grade on the pedals, or information about the driver. It is in this last subgroup where it is required the use of a facial recognition system to get this information which can be used by the multiagent system to define more complex situations and therefore, more precises.



Figura 15: *Kinect* v2.0 camera.

To carry out this task, it will be used the device *Kinect* v2.0 (Figure 1.3 and it will be implemented an application using the tools supplied by the manufacturer. The system developed in this project will be limited to monitor the driver and send the information to the computer where it is executed the driving simulator. The basic information is detected by the device *Kinect* and, besides, it is implemented an eye detected method no including among the functionality of the device. In addition, it will be used an alarm system implemented before in an end-of-master project [33] adding new own alarms, which will bring more information to the multiagent system. This system is dedicated to the merger and data analysis, determining the dangerousness of traffic situation and making the decision of activating an alarm or not.

In the figure 1.4 they can be observed the components of the global Project in which the system developped in this end-of-degree project will work.



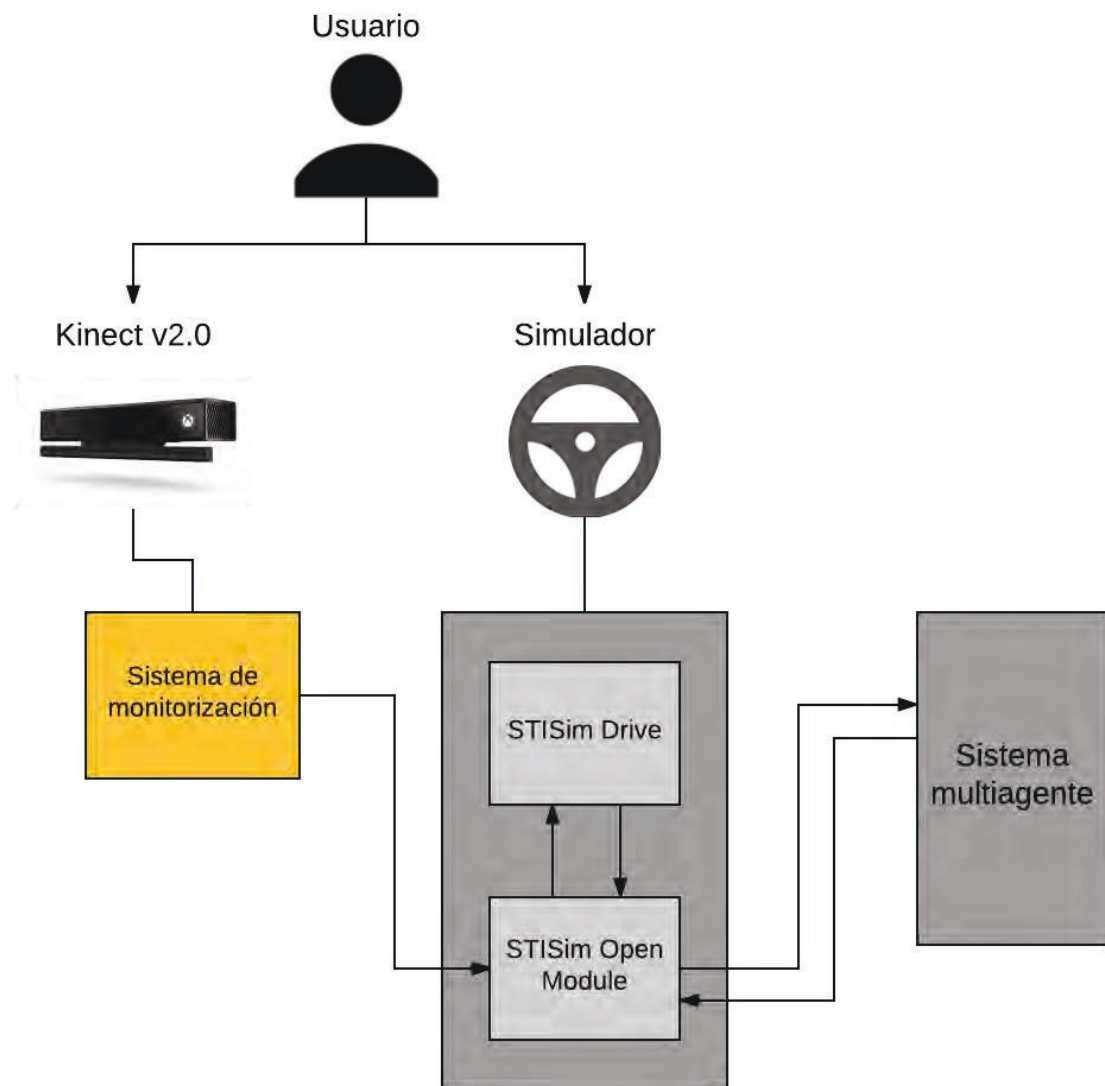


Figura 16: Diagram of the global project where the monitoring system will be used.

Therefore, synthesizing the above, this end-of-degree goal is divided into the following:

- Extracting information from the facial recognition conducted by the Kinect v2.0 device.
- Implementation of a method for detect the position of the pupils.
- Generate auxiliar information using an alarm system.
- Communication of the information in real time between computers.

In this system it can be found five main modules which interact with each other: graphic interface, frame management, eye tracking, alarm system and connection via socket. Each one of these components represent one main functionality of the system. In the figure 4.1 it can be seen how this systems are interconnected.

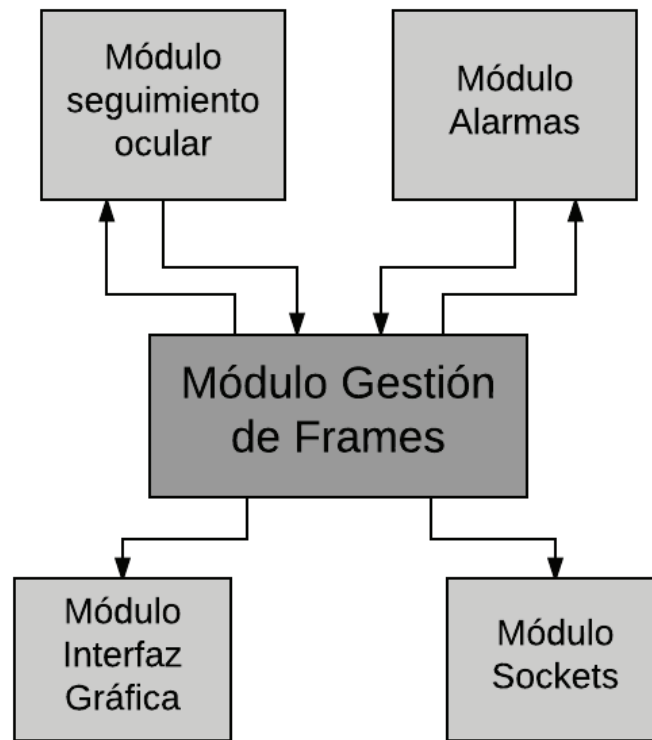


Figura 17: Module diagram of the system's architecture.

As a model, it has been used a project created by the Automatic department for an end-of-master project called *Detección del estado emocional de un conductor mediante el uso de imágenes infrarrojas* [33]. The program operation is divided in several stages which are presented below in the figure 5.1.

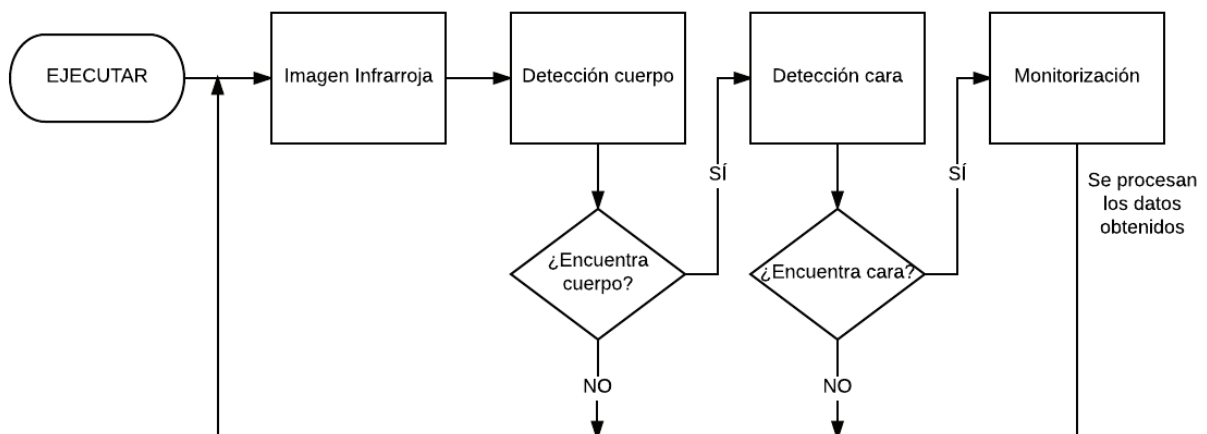


Figura 18: Detection of the user's face process.

Although the driving simulation is executed in an indoor environment and with controlled lighting conditions, it must take into account the conditions that would apply in a real urban environment. Therefore, it is used the infrared camera that Kinect offers, so that lighting is not a problem when performing monitoring. This camera emits an infrared beam and is able to detect its reflection on the objects, thereby solar radiation does not interfere in the process (figure 5.7).



Figura 19: Image captured by the infrared camera.

Using the information captured by the camera it can be shown in the graphic interface the points of interest on the user's face to know their position at all times. Also the status of these points is shown, as shown in the figure 5.9.

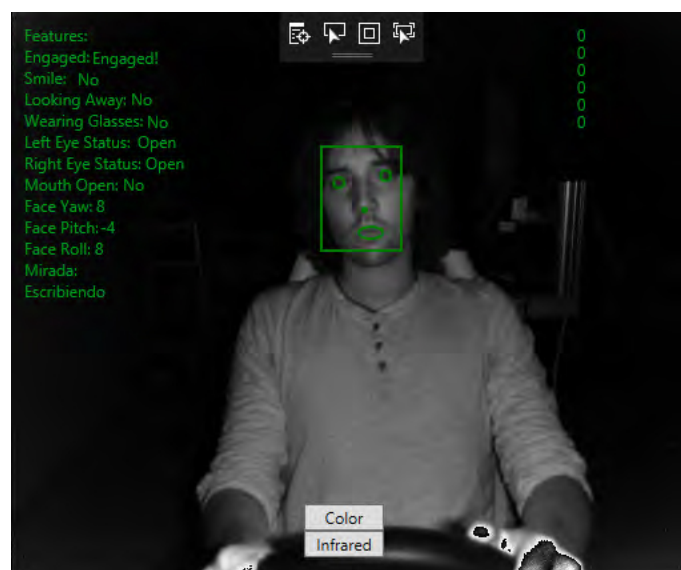


Figura 20: Example of the user's face detected by the infrared camera.

The next aspect to be monitored in the user is the position of the pupils. The camera offers many features that pose a considerable amount of information, but knowing the position of the eyes is considered essential to guarantee the safety of the driver. Without this information, the user could have his head perfectly aligned with the road and the system would consider that there's nothing wrong, while the user may be looking at another direction, creating a dangerous situation.

To determine the position of the pupil, images captured by the high-resolution camera are used (such as the one seen in the figure 5.11)



Figura 21: Image captured by the high-definition camera.

The color camera, as the infrared camera, recognizes a number of points which belong to the human face, but, in this case, by having higher resolution, many more points can be accessed, which can be observed in the figure 5.12 in color blue.



Figura 22: Face points detected by the high-resolution camera.

To detect the position of the pupil in an image captured by the camera, tools provided by the library EmguCV are used and that image is put through a process shown in the figure 5.20.

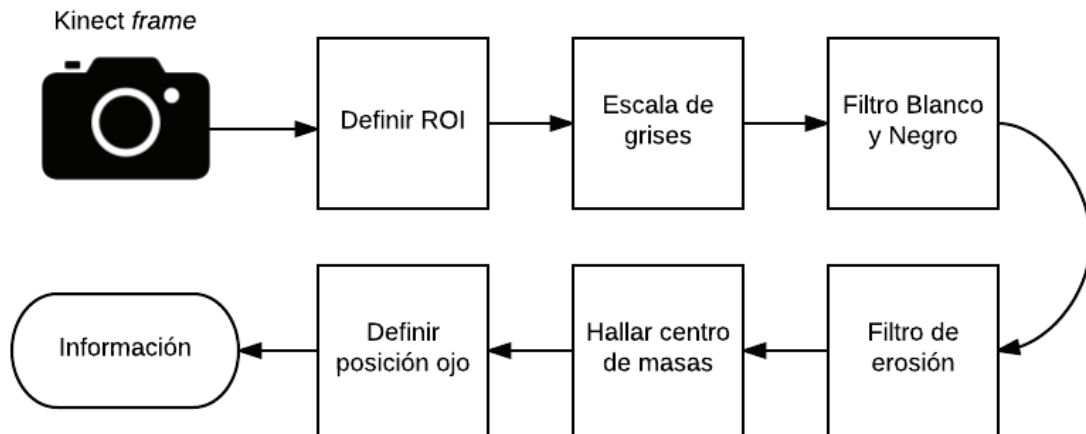


Figura 23: Process to detect the position of the eyes.

To end the monitoring of the user, it is proposed to use the information gathered until now to create new information, that is, interpreting the data gathered in a way that new useful information can be gotten from them. For that an alarm system has been used. The system used has been developed in the end-of-master project *Detección del estado emocional de un conductor mediante el uso de imágenes infrarrojas* [33] and a new case has been added taking into account the position of the eyes.

For this alarm system, the following cases that can generate dangerous situations are considered.

- Long blinks
- Open mouth
- Turn of the head
- Speak
- Drink consumption
- Eyes position

As previously explained, the system is connected with the driving simulator by implementing a synchronous socket, using a client-host architecture. The computer where this application is acts as the server and activates a socket while waiting for connection. Once the socket is open and the server waits for a connection, the simulator can be executed in the computer that acts as server. In this moment, the client sends a message to the host requesting information about the monitoring of the user. When this message is received, the server sends to the client one line of text which contains the information gathered by the program explained above.

To evaluate the performance of the application ten test subjects of different genders and different physical appearance have been chosen, so the system actions are observed by monitoring different subjects with different features. In the figure 24 images of the test subjects can be seen.

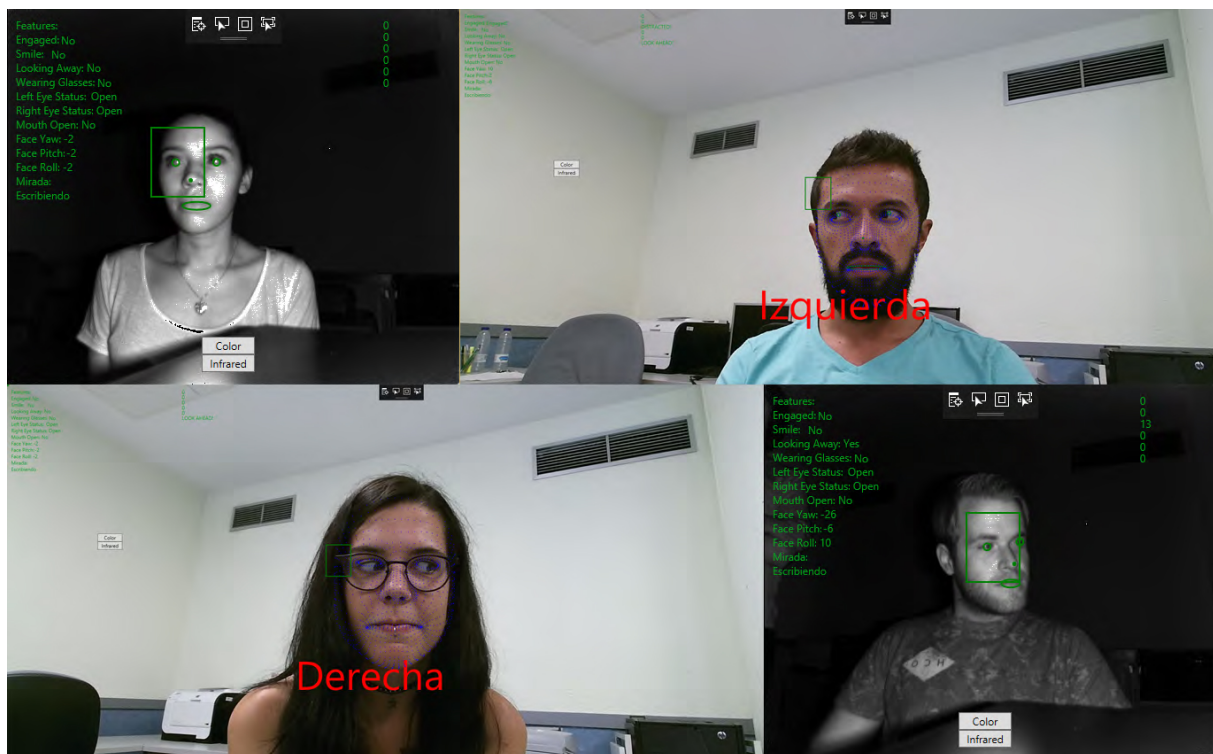


Figura 24: Images of the test subjects.

To end, the reached conclusions are exposed after the implementation and development of the end-of-degree project, as well as future lines of work to which this project can be directed.

Once project is completed, it can be concluded that the main objectives have been accomplished. It has been achieved the user monitoring realization and the control at all times of the position and situation of different facial features to consider. It has been found a way to collect this information and show it by screen. It has been always got a reliable method to carry out the recognition of these characteristics regardless of lighting conditions using an infrared camera. Moreover, it has been generated additional information based on the initial data through the alarm system.

It has been also implemented the recognition system of the position of the user's pupils, characteristic that the Kinect v2.0 device is not able to detect. This new functionality can be used in propitious lighting conditions.

Another goal of this project was to establish communication between the system which control the monitoring and the system where is run the driving simulation. This goal has also fulfilled by implementing a synchronous socket which connects the computers which host these systems and allow the exchange of messages among them.

During the realization of this project it has become aware of the magnitude and difficulty of a project of this size, having to set the main objectives and planning design times, implementation, etc. It has also been made contact with the work carried out by the investigation group CAOS of this university, helping to develop a small part of a much larger project. During the process, certain job and learning skills needed for the future have been acquired.

Also worth mentioning the knowledge gained about the users monitoring and the motion capture by devices as it has been used during the project, learning to cope with tools needed for its handling, as well as the knowledge about image processing in system which works in real time.

In general technical conclusion, they have been applied satisfactorily knowledge gained during the degree to a problema on computer engineering.

Related to future works, the system developped in this end-of-degree project is a first approach to a vehicle driver monitoring and it is only checked raw data about the physical state of the user. In the future, it can go deeper into the area of emotions, which, as already mentioned, have a very big impact on the way to behave people and can influence detrimentally in driving. By a further face detection, these driving emotions could be considered. Probably, it could be used the information captured by the high definition camera of Kinect v2.0 device, although it is possible that facial expressions are not detailed enough.



The next step of this Project would be the integration of the information captured by this system into multiagent system developed by CAOS department. This driver information will have to be used within the ontology of the multiagent system and so, this will be able to consider much more complex situations during the driving simulation.

The other aspect to improve is monitoring the position of the pupils, this is controlled independently of the head position and they are only considered three values for eye position. As future work, it is proposed to find another method which allows knowing the pupil position accurately and the look direction considering head position. This method, also, to be even more precise will have to work with infrared images, to not depend on lighting and to be used in any situation while driving.



# Capítulo 1

## Introducción

La seguridad en las carreteras ha sido siempre una tema de gran importancia en la sociedad y en los últimos años se han observado importantes avances tecnológicos para garantizarla. Los accidentes de tráfico son la mayor causa de muerte entre las personas de entre 15 y 29 años y en la actualidad se ha llegado a la cifra de 1,25 millones de muertes por año debido a colisiones de tráfico [1]. El número de muertes ha comenzado a estabilizarse desde 2007 a causa de los avances que han permitido mejorar la seguridad vial (Figura 1.1), es decir, los Sistemas Avanzados de Asistencia a la Conducción o ADAS (*Advanced Driver Assistance Systems*).

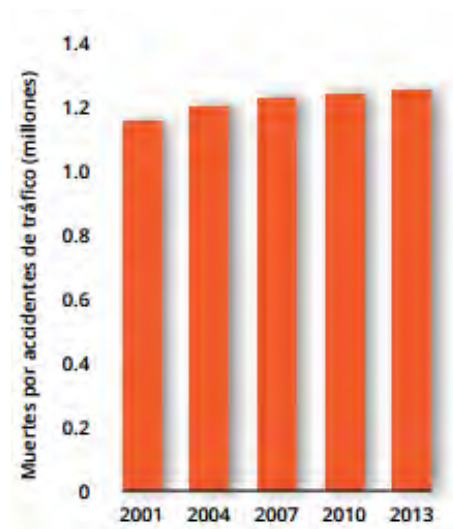


Figura 1.1: Muertes por accidentes en todo el mundo en los ultimos años (Fuente: [www.who.int](http://www.who.int)).

Por este motivo, hay que tener en cuenta la importancia de estos sistemas, que no sólo ayudan a las personas dentro del vehículo, si no que aumentan la seguridad general en la carretera, ya que la mitad de las víctimas de los accidentes mortales son peatones, ciclistas o motoristas, los cuales no cuentan con sistemas de seguridad tan sofisticados como los que puede incorporar un coche.

Seguir invirtiendo en este sector puede llegar a salvar muchas vidas, además de evitar pérdidas económicas relativamente altas ocasionadas por los accidentes. Entre los objetivos de la Organización Mundial de la Salud (OMS) para el año 2020 está el reducir en un 50 % el número de muertes y traumatismos por accidente de tráfico.

## 1.1. Motivación y objetivos

Gran parte de los accidentes de tráfico son ocasionados por distracciones del conductor o por falta de atención, las cuales aumentan con el grado de fatiga o el no prestar atención únicamente a la conducción. La mayoría de sistemas de ayuda a la conducción pueden evitar accidentes en estas situaciones de falta de atención, pero, dado que del 20 % al 30 % de los accidentes de tráfico se relacionan con el cansancio [2], se hace necesaria la existencia de sistemas que monitoricen el comportamiento del conductor y eviten o adviertan de este estado de fatiga. De esta manera no se deja la seguridad del coche únicamente en manos de estos sistemas que evitan los accidentes en el momento en el que van a ocurrir. Es necesario, además, detectar y avisar al conductor de las situaciones de peligro causadas por la fatiga. Lo más común es que, el conductor, aunque es consciente de su cansancio, infravalora el riesgo de quedarse dormido al volante, pudiendo causar un accidente de gran gravedad en caso de cerrar los ojos durante unos segundos [3].

La fatiga no es el único factor a tener en cuenta, como se ha dicho antes, también pueden existir distracciones por llevar a cabo otras actividades mientras se conduce o por falta de atención debido a las múltiples emociones que puede experimentar un ser humano. Nuevamente, el conductor es el punto central al que hay que prestar atención para determinar en qué estado se encuentra y poder evitar situaciones peligrosas.

En este trabajo fin de grado se realizará una primera aproximación a un sistema de monitorización del usuario, el cual podrá ser incorporado más adelante a un sistema de asistencia a la conducción completo.

## 1.2. Descripción del problema

La necesidad de un sistema de reconocimiento facial comienza con un Sistema Avanzado de Asistencia a la Conducción utilizado por el grupo de investigación *CAOS* (Control, Aprendizaje y Optimización de Sistemas) de la Universidad Carlos III de Madrid [4], para el cual se está desarrollando un sistema basado en agentes con un alto nivel de razonamiento. Este sistema se utiliza para realizar simulaciones de conducción en un entorno urbano poniendo especial atención en la seguridad vial. El simulador utilizado es *STISIM Drive* y se puede ver en la Figura 1.2 [6].



Figura 1.2: Plataforma de simulación basada en el simulador *STISIM Drive*.

En este sistema multiagente, como su propio nombre indica, existen varios agentes con diferentes roles, los cuales se pueden separar en dos grupos. Por un lado, los que tienen como objetivo la recogida de información y por otro, los utilizados para el análisis y fusión de esta información para definir qué situaciones dentro de la simulación pueden resultar peligrosas. Dentro del primer grupo se puede recoger información del entorno de simulación, como peatones o señales de tráfico, información del vehículo, como la velocidad o el grado de presión en los pedales, o información acerca del conductor. Es en este último subgrupo donde se requiere el uso de un sistema de reconocimiento facial para recoger esta información que pueda utilizar el sistema multiagente para conseguir definir situaciones más complejas y por lo tanto, más precisas.



Figura 1.3: Cámara *Kinect* v2.0.

Para llevar a cabo esta tarea se utilizará el dispositivo *Kinect* v2.0 (Figura 1.3) y se implementará una aplicación utilizando las herramientas proporcionadas por el fabricante del mismo. Por lo tanto, el problema planteado en este trabajo fin de grado es la implementación de las funcionalidades necesarias para la recogida de información y el envío de ésta al equipo donde se ejecuta el simulador *STISIM Drive*. Se trabajará, por un lado, con

la información proporcionada por el propio dispositivo (posición de la cabeza, ojos, etc), con información obtenida de las imágenes captadas por la cámara pero que el dispositivo no proporciona (como la posición de las pupilas) y finalmente, con la información resultante de la interpretación de los datos anteriores en forma de alarmas, proporcionando así más datos acerca del comportamiento del conductor al sistema multiagente. Este sistema se dedica a la fusión y análisis de los datos, tras lo que determina la peligrosidad de la situación de tráfico y toma la decisión de si activar o no una alarma.

En la figura 1.4 se pueden observar los componentes del proyecto global dentro del que funcionará el sistema desarrollado en este trabajo fin de grado.

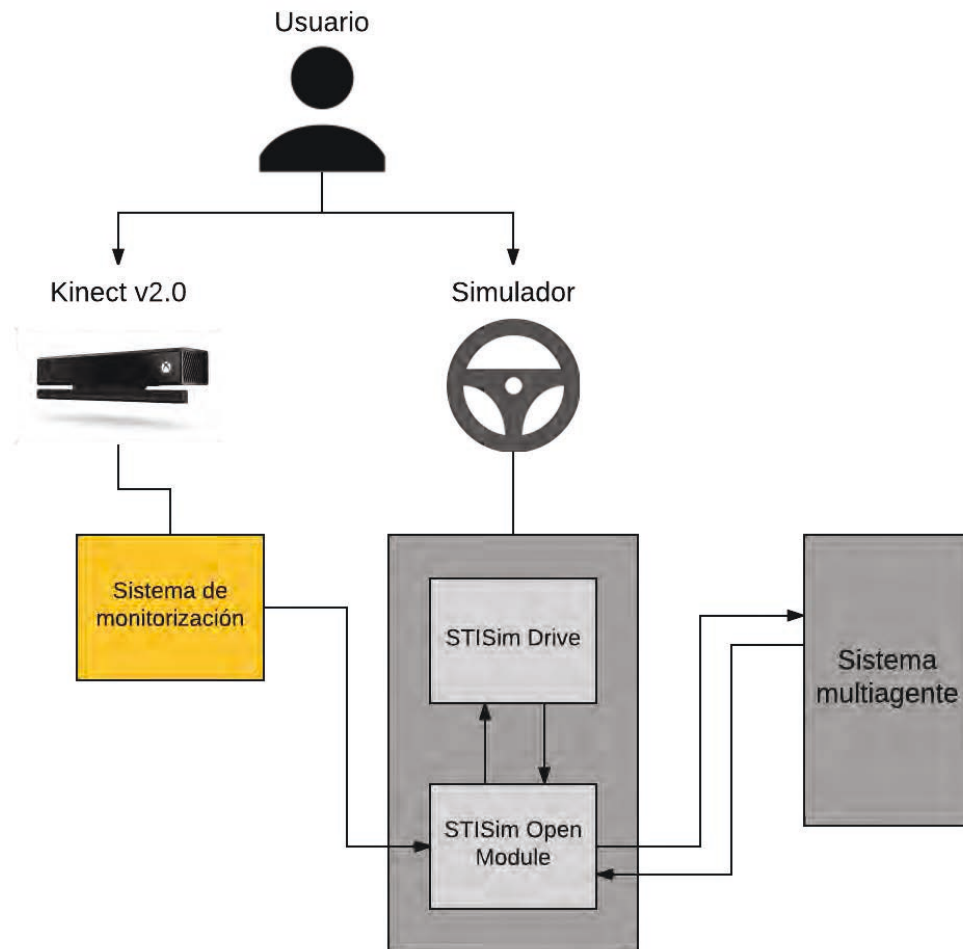


Figura 1.4: Diagrama del proyecto global donde se utilizará el sistema de monitorización.

El sistema desarrollado en este proyecto se limitará a monitorizar al conductor y enviar la información al equipo donde se ejecuta el simulador de conducción. La información básica es detectada por el dispositivo *Kinect* y, como añadido, se implementa un método de detección ocular no incluido entre las funcionalidades del dispositivo. Además se utilizará un sistema de alarmas implementado anteriormente en un trabajo fin de máster [33] añadiendo nuevas alarmas propias.

### 1.3. Alcance del proyecto

Como se ha mencionado anteriormente, en este trabajo fin de grado se persigue un objetivo: la obtención de información mediante la monitorización del usuario utilizando el dispositivo Kinect. Esta información será enviada al equipo donde se ejecuta el simulador de conducción junto con el sistema multiagente.

La información se obtiene de tres fuentes. Primero, se cuenta con las funcionalidades del dispositivo Kinect v2.0 para obtener los datos relativos a la posición de la cabeza (ángulos de giro) y diversas expresiones faciales (ojos cerrados, sonreír, boca abierta, etc). La segunda fuente de información ha tenido que ser implementada, ya que se trata de la posición de las pupilas y la cámara Kinect no proporciona esa información. Para conseguir esto se ha desarrollado un método que permite detectar la posición de los ojos utilizando la librería para procesamiento de imágenes *EmguCV*, que permite utilizar la librería *OpenCV* en lenguajes compatibles con .NET como C#, que es el utilizado en este proyecto.

Para finalizar, la tercera fuente de información será un número diverso de alarmas que se activarán en función de la información obtenida de las dos fuentes anteriores. De esta forma, se pueden transformar los datos “en bruto” de las dos primeras fuentes en nuevos datos que pueden aportar información nueva y más precisa al sistema multiagente. Es decir, existen datos que en un determinado momento pueden no ser de utilidad si no se integran con el resto de información a la que tiene acceso el sistema multiagente, como “*ojo derecho cerrado*” o “*boca abierta*” y otros como “*hablando*” o “*distraído*” aportan información relevante sobre el estado del conductor sin necesidad de más información del exterior para completarlos.

Para completar el trabajo es necesario implementar la comunicación entre los dos equipos donde se ejecuta el simulador de conducción y la monitorización del conductor, actuando estos como cliente y servidor respectivamente. Toda la información recogida por la cámara en cada instante de tiempo es estructurada en una manera concreta y es enviada al equipo que actúa como cliente para poder ser utilizada en el sistema multiagente. Para conseguir esto se ha implementado un método de comunicación basado en sockets para conectar ambos equipos.

Por lo tanto, sintetizando lo anterior, el objetivo de este trabajo fin de grado se divide en los siguientes :

- Extracción de la información obtenida del reconocimiento facial llevado a cabo por el dispositivo Kinect v2.0.
- Implementación de un método para detectar posición de las pupilas.
- Generar información auxiliar mediante un sistema de alarmas.
- Comunicación de la información en tiempo real entre equipos.

## 1.4. Restricciones

En este apartado se definen las restricciones aplicadas al proyecto, es decir, se describen cuales son los requisitos mínimos para su correcto funcionamiento. Estas restricciones se dividen en software y hardware

A continuación se describen las restricciones relativas al software:

- El equipo en el que se conecta el simulador debe de contar con el software *STISIM Drive 3.03.08* o versiones superiores.
- Para ejecutar el proyecto desarrollado en *C#*, se necesita tener el *SDK* adecuado.

A continuación se describen las restricciones relativas al hardware:

- Se necesitan tres monitores para simular el entorno visual del conductor.
- El dispositivo de grabación utilizado para la detección facial debe ser una *Kinect v2.0*.
- Se necesitan volante, pedales, palanca de cambios y asiento para simular la conducción.
- Es necesario establecer una conexión entre los dispositivos para que estos se comuniquen entre ellos.

## 1.5. Marco regulador

El proyecto se basa en la monitorización del conductor del vehículo, por lo que, al grabar el rostro del usuario, se están utilizando datos personales del mismo. Por lo tanto será necesario cumplir con la Ley Orgánica 15/1999 de Protección de Datos de Carácter Personal [24].

Además, el sistema se crea con la finalidad de ser utilizado en el futuro dentro de un vehículo real y podría tomar el control de éste si fuera necesario. Esto hace necesario conseguir la autorización de pruebas o ensayos de investigación realizados con vehículos de conducción automatizada en vías abiertas al tráfico en general, la cual se encuentra en la Instrucción 15/V-113 de la DGT [22] [23].

## 1.6. Entorno operacional

En esta sección se exponen los elementos software y hardware que han sido utilizados durante el desarrollo del proyecto.

El software utilizado ha sido:

- Entorno de desarrollo *Visual Studio Community 2015*
- *Kinect for Windows Software Development Kit (SDK) 2.0*
- Sistema de composición de textos *LaTeX*
- Simulador de conducción *STISIM Drive M100 Series* [6]
- Librerías de procesamiento de imagen *EmguCV 3.1.0*
- Sistema de grabación *Camtasia*
- Herramienta *Lucidchart*

El hardware utilizado ha sido:

- Equipo que ejecuta el simulador
  - Procesador Intel(R) Core(TM) i7-3770 CPU @ 3.4GHz
  - Memoria RAM de 16GB
  - SO Windows 7 Professional de 64 bits Service Pack 1
- Equipo que ejecuta el proyecto
  - Procesador Intel(R) Core(TM) i7-4790K CPU @ 4.00GHz
  - Memoria RAM de 16GB
  - SO Windows 8.1 Professional de 64 bits
- Juego de volante de cuero, pedales y cambio de marchas Logitech G27

## 1.7. Organización de la memoria

A continuación se describe la estructura del documento, mediante una explicación breve de cada capítulo que lo compone.

En el primer capítulo se realiza una introducción del proyecto. Para ello se exponen las motivaciones y objetivos, la descripción del problema a tratar, el alcance del proyecto, restricciones, marco regulador y entorno operacional del proyecto. Se finaliza la introducción con este mismo apartado, describiendo la estructura del documento.

En el segundo capítulo se expone el estado del arte de temas relacionados con el ámbito de los Sistemas de Ayuda a la Conducción y modelado y detección de emociones.

En el tercer capítulo se describe el análisis del proyecto, representando todos los requisitos funcionales y no funcionales y los casos de uso considerados.

En el cuarto capítulo se muestra la arquitectura que presenta el sistema y se describe cada módulo de manera independiente y su interacción entre ellos.

En el quinto capítulo se describe el proceso de implementación de todas las funcionalidades del sistema.

En el sexto capítulo se presenta la experimentación realizada para evaluar el rendimiento del sistema y el cumplimiento de los objetivos marcados.

En el séptimo capítulo se describe la metodología utilizada para desarrollar el proyecto, así como la planificación y presupuestos. También se analiza el impacto que puede ocasionar el proyecto en la sociedad o en el medio ambiente.

En el octavo capítulo se exponen las conclusiones finales tras la realización del proyecto, divididas en conclusiones técnicas y personales. También se analizan los trabajos futuros que se pueden realizar partiendo del trabajo realizado.



# Capítulo 2

## Estado del arte

En este capítulo se tratan diversas temáticas relacionadas con el objetivo de este trabajo fin de grado. Para empezar se describen varios ejemplos existentes de *Sistemas Avanzados de Asistencia a la Conducción (ADAS)* que se pueden encontrar hoy en día en nuestra sociedad. Se hace una distinción entre dos tipos de sistemas de asistencia: los que ejercen control sobre el vehículo y los que simplemente informan de ciertos eventos mediante el uso de indicadores o alarmas. Más adelante se exponen diferentes formas de catalogar o entender las emociones, lo cual puede ayudar a entender las diferentes emociones que se pueden manifestar durante la conducción. A continuación se describe un proyecto conocido como *ZenCar* y se explicará cómo poder detectar las diferentes emociones que puede experimentar una persona al volante y cuál es su impacto en la forma de conducir. Para finalizar se expondrán las características del dispositivo Kinect v2.0, el cual se va a utilizar en este trabajo fin de grado, especificando sus características y funcionalidades.

### 2.1. Sistemas Asistencia a la Conducción

Los Sistemas Avanzados de Asistencia a la Conducción o ADAS (*Advanced Driver Assistance Systems*) se integran en los vehículos para ayudar al conductor durante el trayecto de diversas formas. Existen diferentes tipos de sistemas de asistencia, los cuales cuentan con multitud de sensores para recopilar la información necesaria para su funcionamiento. Estos sistemas aumentan la seguridad durante la conducción, ayudando a evitar situaciones de riesgo que pudieran acabar en colisiones de tráfico o, simplemente, aumentando la comodidad a la hora de conducir y mejorando dicha experiencia. En los últimos años se ha podido observar un gran avance en el funcionamiento de estos sistemas, los cuales cada vez son más fiables.

Como ya se ha explicado, en este capítulo se diferencia entre los sistemas que toman el control del vehículo en caso de peligro y los sistemas meramente informativos. Los primeros controlan el funcionamiento de ciertas funcionalidades del vehículo para garantizar la seguridad del conductor en situaciones en las que éste no puede reaccionar. El segundo tipo de sistemas son los que ofrecen ayuda informativa mediante alarmas o señales, siendo el conductor el encargado de controlar el vehículo en todo momento.

### 2.1.1. Sistemas de asistencia con control sobre el vehículo

#### Control de velocidad de crucero adaptativo (ACC)

Es un sistema que tiene en cuenta las condiciones del tráfico para mantener una velocidad adecuada a las condiciones de la carretera. Este sistema utiliza una serie de radares que, en caso de detectar otro coche en el mismo carril a una velocidad inferior a la propia, toma control de los frenos para regular la velocidad y mantener una distancia de seguridad determinada. En el momento que el carril queda libre, el sistema vuelve a acelerar para recuperar la velocidad de crucero prefijada [7] (Figura 2.1).



Figura 2.1: Control de crucero adaptativo (Fuente: [www.motor.mapfre.es](http://www.motor.mapfre.es)).

#### Sistema de frenada de emergencia autónoma (AEB)

Es un sistema inteligente de seguridad que, además de aumentar la presión de frenado, actúa con independencia para evitar una colisión con otro vehículo, persona, obstáculo además de otros peligros. El sistema está pensado para ser utilizado en entornos urbanos a una velocidad no superior a los 30 km/h. Básicamente utiliza radares para controlar el entorno inmediato (a máximo 10 metros por delante del vehículo) y evitar una colisión por un despiste del conductor. Si éste realiza alguna acción para evitar la colisión, como frenar, el sistema se desactiva [8] (Figura 2.2).

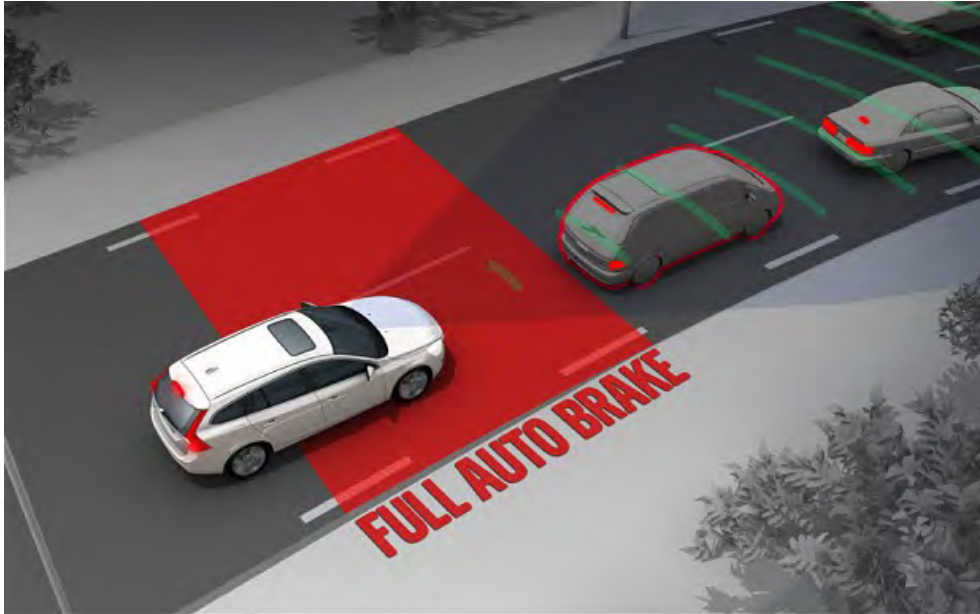


Figura 2.2: Sistema de frenado de emergencia (Fuente: [www.telegraph.co.uk](http://www.telegraph.co.uk)).

### Asistente de mantenimiento en el carril

Este sistema reconoce las líneas que delimitan el carril mediante una cámara situada sobre el parabrisas y, en caso de que el vehículo sobrepase o pise estas líneas de manera involuntaria, actúa sobre la dirección del vehículo para mantener la trayectoria. Existen sistemas más sencillos, los cuales simplemente alertan al conductor mediante alarmas sonoras, mensajes o vibración del volante o asiento [9] (Figura 2.3).

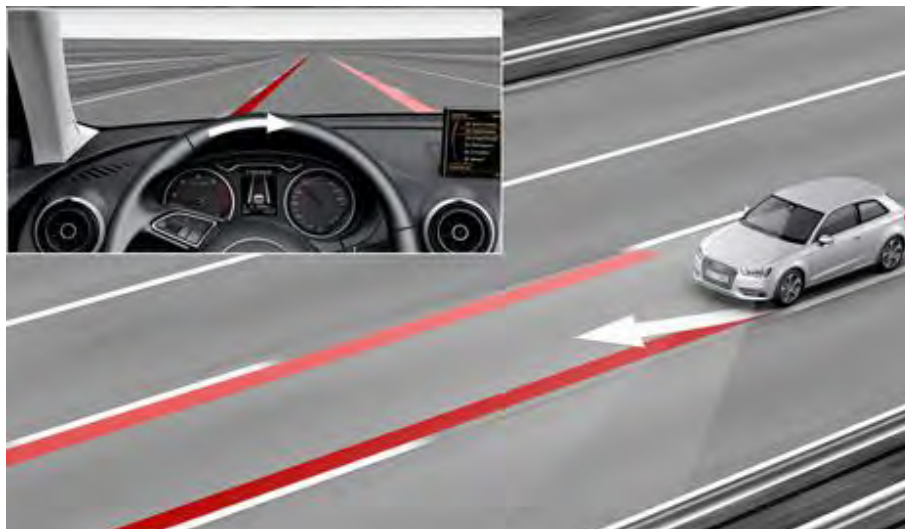


Figura 2.3: Asistente de mantenimiento en el carril (Fuente: [www.motor.mapfre.es](http://www.motor.mapfre.es)).

## Asistente para atascos de tráfico

Este sistema combina los tres sistemas anteriores, de forma que el coche pueda seguir al vehículo que esté delante en una situación de atasco. El coche se mantendrá en todo momento a la distancia de seguridad especificada y frenará y acelerará en caso de ser necesario. Todo esto mientras el control de mantenimiento de carril controla que el vehículo se mantenga dentro del carril. Con este sistema básicamente el coche se conduce solo durante un atasco de tráfico. [10] (Figura 2.4).



Figura 2.4: Asistente para atascos de tráfico (Fuente: [www.xataka.com](http://www.xataka.com)).

## Sistema de aparcamiento automático

Es un sistema capaz de realizar la maniobra de aparcamiento. Los primeros sistemas no eran tan complejos como los actuales, teniendo que realizar el usuario ciertas acciones como el cambio de marchas o acelerar y frenar, pero los sistemas actuales pueden llegar a realizar la maniobra de aparcamiento completa mediante sensores de proximidad y la electrónica del vehículo [11] (Figura 2.5).

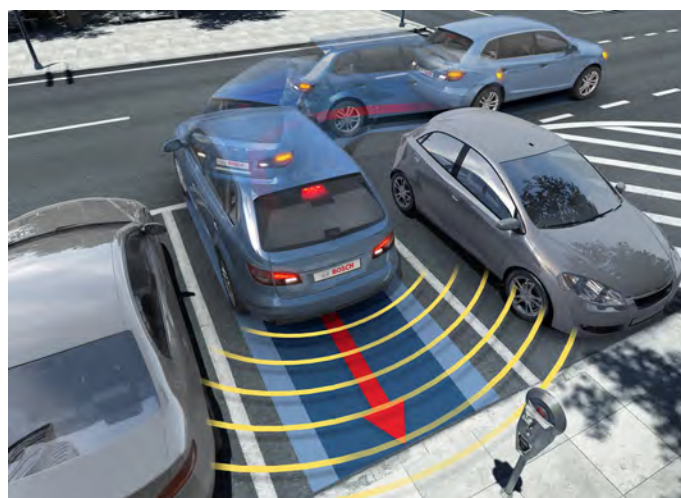


Figura 2.5: Sistema de aparcamiento automático (Fuente: [www.diariomotor.com](http://www.diariomotor.com)).

## Sistema de iluminación adaptativa

Es un sistema que permite una iluminación automática de la carretera sin perjudicar al resto de conductores. Existen varios ejemplos de esta tecnología, como las luces dinámicas en curva, las cuales giran de forma coordinada con la dirección del coche, de manera que la luz alumbre la trayectoria de la carretera (Figura 2.6).

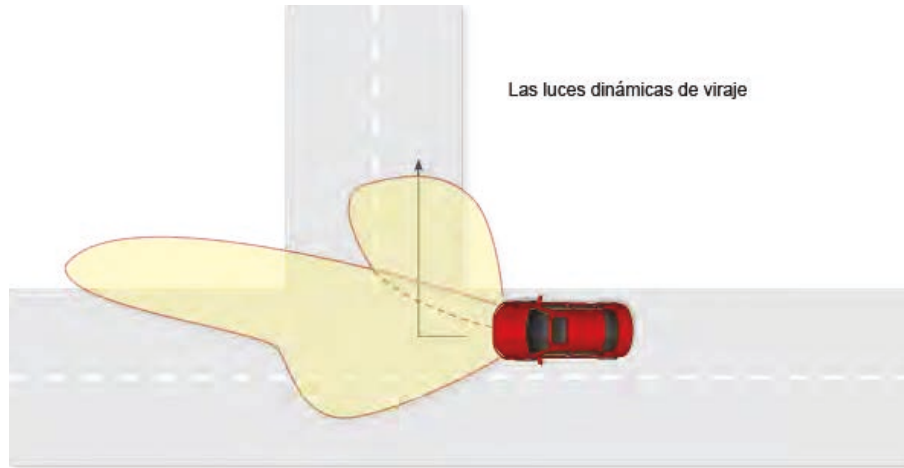


Figura 2.6: Luces dinámicas en curva (Fuente: [www.car-lighting-selector.com](http://www.car-lighting-selector.com)).

Otro ejemplo es el asistente de luces de carretera, el cual se encarga de realizar el cambio entre las luces de cruce y carretera según las necesidades, intentando activar las luces de carretera (largas) siempre que sea posible para circular con mayor seguridad y no deslumbrar al resto de conductores [12] (Figura 2.7).

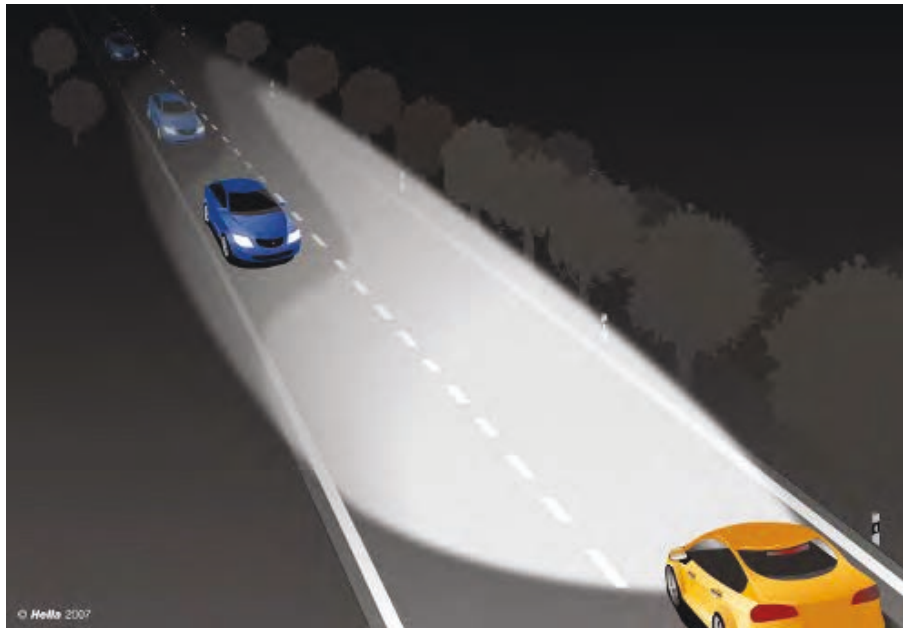


Figura 2.7: Asistente de luces en carretera (Fuente: [www.circulaseguro.com](http://www.circulaseguro.com)).



## 2.1.2. Sistemas de asistencia informativos

### Detector de ángulo muerto

Este sistema controla las zonas que el conductor no puede ver mediante los espejos retrovisores. Puede funcionar mediante radares, sensores o cámaras de vídeo para advertir al conductor de la presencia de vehículos en estas zonas no accesibles por la vista con el fin de evitar situaciones peligrosas. El sistema alerta al conductor mediante alarmas acústicas, luminosas o incluso por vibración [13] (Figura 2.8).

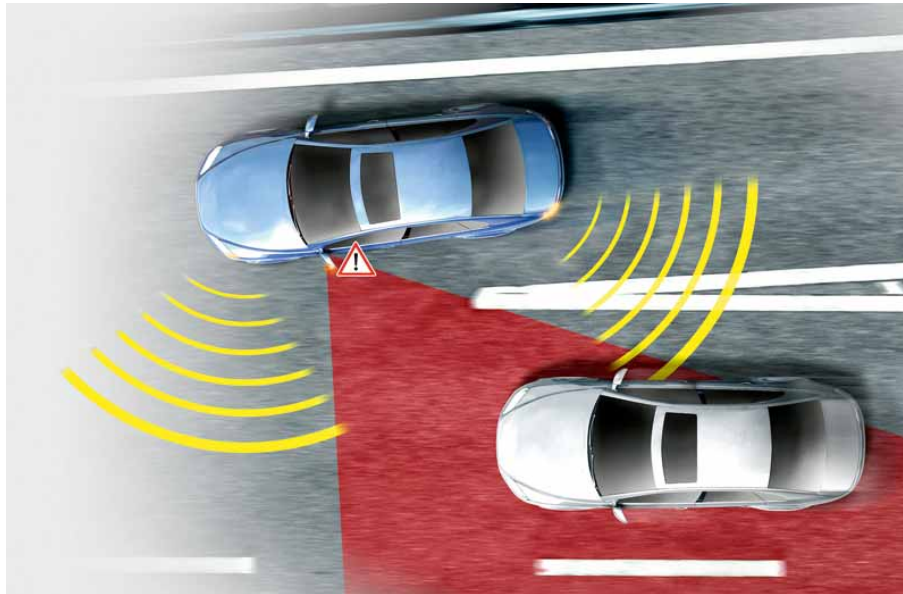


Figura 2.8: Detección de ángulo muerto (Fuente: [www.autopista.es](http://www.autopista.es)).

### Asistencia de visión nocturna

Este sistema permite, mediante cámaras termográficas o de infrarrojos, detectar objetos, personas o cualquier obstáculo que escapa al campo de visión iluminado por las luces. La imagen captada por la cámara es mostrada en una pantalla situada en el interior del vehículo, de esta forma se puede reaccionar antes a ciertos peligros. La situación ideal sería que la información de los objetos encontrados fuera proyectada en el parabrisas, evitando así el alternar la vista entre la carretera y la pantalla, lo que puede ocasionar fatiga ocular o un despiste [14] (Figura 2.9).



Figura 2.9: Asistencia de visión nocturna (Fuente: [www.circulaseguro.com](http://www.circulaseguro.com)).

### Sistema de reconocimiento de señales de tráfico

Este sistema, como su propio nombre indica, reconoce las señales de tráfico como las de prohibición, límites de velocidad, etc. Ayuda al conductor a mantener la atención sin que se le escape ninguna limitación señalizada en la carretera. Su funcionamiento es posible mediante cámaras de vídeo que muestran la información en tiempo real, de manera que, si el conductor se despista, puede consultar las limitaciones del tramo actual en el que se encuentra [15] (Figura 2.10).

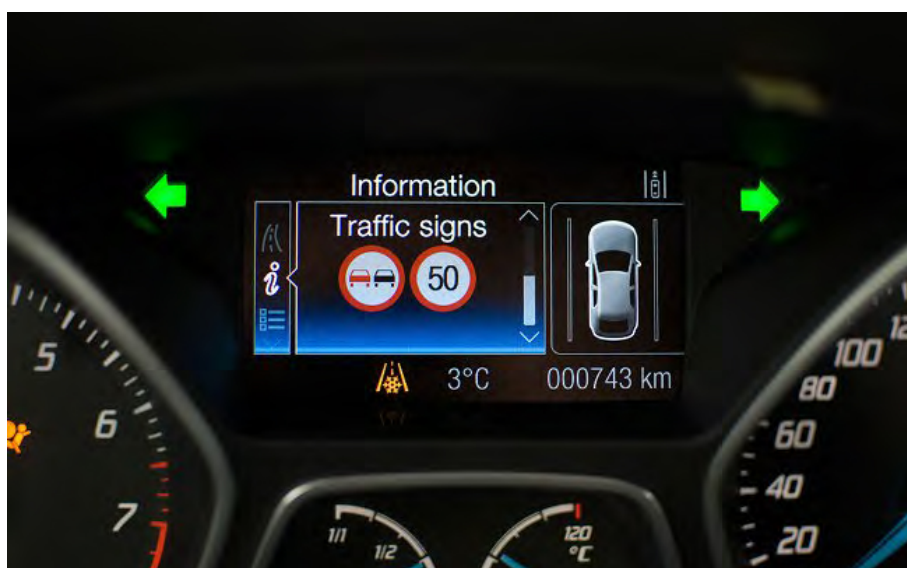


Figura 2.10: Reconocimiento de señales (Fuente: [www.circulaseguro.com](http://www.circulaseguro.com)).

## Sistema de detección de fatiga y concentración al volante

Este sistema es capaz de detectar si el conductor presenta un estado de fatiga o cansancio destacable o si se está quedando dormido mediante el uso de diferentes sensores y advertirle para así evitar un posible accidente. Un ejemplo de los sensores utilizados son los sensores en el volante, mediante los cuales se pueden detectar anomalías en la forma en la que el usuario está conduciendo. Al conducir normalmente se suelen hacer ligeras correcciones en la posición del volante para mantenerse dentro del carril, mientras que, en condiciones de cansancio, estas correcciones se vuelven bruscas si los ojos del conductor se cierran. En este momento el sistema activa una alarma sonora y visual en el cuadro de instrumentos para alertarnos de la situación de fatiga (Figura 2.11).



Figura 2.11: Detección de fatiga (Fuente: [www.telematicsnews.info](http://www.telematicsnews.info)).

Otro sensor más avanzado que se utiliza en estos sistemas es una cámara con un sistema de reconocimiento facial. Esta cámara supervisa los ojos del conductor e indica si el parpadeo de estos indica sueño. También controla la mirada del conductor, determinando en qué dirección está mirando, de esta manera se puede determinar si el usuario no está atendiendo a la carretera por cualquier motivo. Además de los ojos, este sistema realiza un reconocimiento facial completo, pudiendo reconocer actividades como bostezos u otras expresiones faciales que indiquen fatiga. También se pueden reconocer otros estados de ánimo como el nerviosismo o el estrés. En estos casos se pone en conocimiento de la situación al conductor mediante una alarma como ya hemos mencionado antes [16] (Figura 2.12)



Figura 2.12: Detección de fatiga (Fuente: [www.motorpasionfuturo.com](http://www.motorpasionfuturo.com)).



## 2.2. Modelado de Emociones en el usuario

Daniel Goleman empezó a utilizar el término “inteligencia emocional” para referirse a los beneficios que supone poder conocer e identificar las emociones humanas y controlar las formas de expresarlas. También utiliza el término para definir la capacidad de automotivación y desarrollar empatía para relacionarse con emociones ajenas. De esta manera se puede aumentar el bienestar tanto psíquico como físico de una persona [17].

Las emociones son estados mentales que producen cambios fisiológicos en el cuerpo humano. Han existido varias teorías sobre cómo clasificar o reconocer las emociones, entre ellas las que se muestran a continuación.

### 2.2.1. Modelo Plutchik

Robert Plutchik realizó una división del conjunto de las emociones en ocho categorías básicas o primarias que hacen referencia a funciones específicas relativas a la supervivencia: alegría, confianza, miedo, sorpresa, tristeza, asco, ira y anticipación. Combinando estas emociones primarias entre ellas se obtienen el resto de emociones para completar el abanico. Esta teoría defiende que las emociones tienen un grado de intensidad que puede variar de unas a otras, es decir, una emoción como la ira representa una intensidad mayor que el enfado, pero menor que la furia. El grado de intensidad de la emoción está relacionado con el grado en que se motiva una conducta afín a ella.

Plutchik representó su clasificación de las emociones de forma gráfica mediante una rueda compuesta por ocho emociones básicas (alegría, confianza, miedo, sorpresa, tristeza, asco, ira y anticipación) y ocho emociones avanzadas que se obtienen de la combinación por pares de las anteriores (amor, sumisión, pavor, decepción, remordimiento, desprecio, agresividad y optimismo). Estas emociones avanzadas se pueden obtener combinando diadas primarias, secundarias o terciarias, dando lugar a emociones menos frecuentes [18] (Figura 2.13).

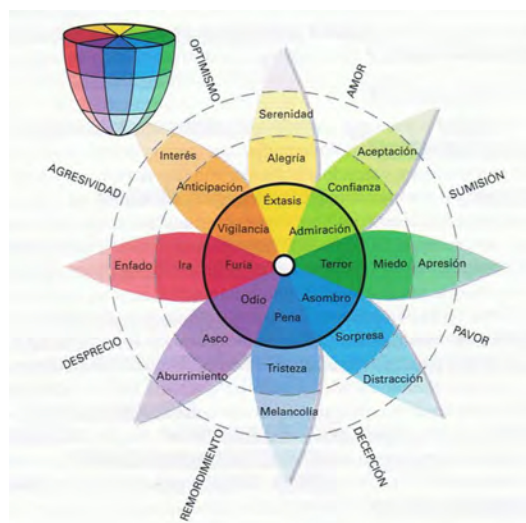


Figura 2.13: Modelo de Plutchik (Fuente: [www.vivirmejorconmigo.com](http://www.vivirmejorconmigo.com)).

### 2.2.2. Modelo de Russell

James Russell diseñó el modelo circular del sistema afectivo. Este sistema está compuesto por dos variables ortogonales que definen un plano cartesiano. La variable horizontal indica el nivel de activación (a la derecha las emociones que denotan mayor excitación y a la izquierda las que denotan relajación) y la vertical el nivel de placer (arriba emociones que representan agrado y abajo las que representan desagrado). En este plano se colocan catorce ejes que representan emociones opuestas y se utilizan colores en la parte central del diseño empleando el mismo criterio mediante colores complementarios [19] (Figura 2.14).

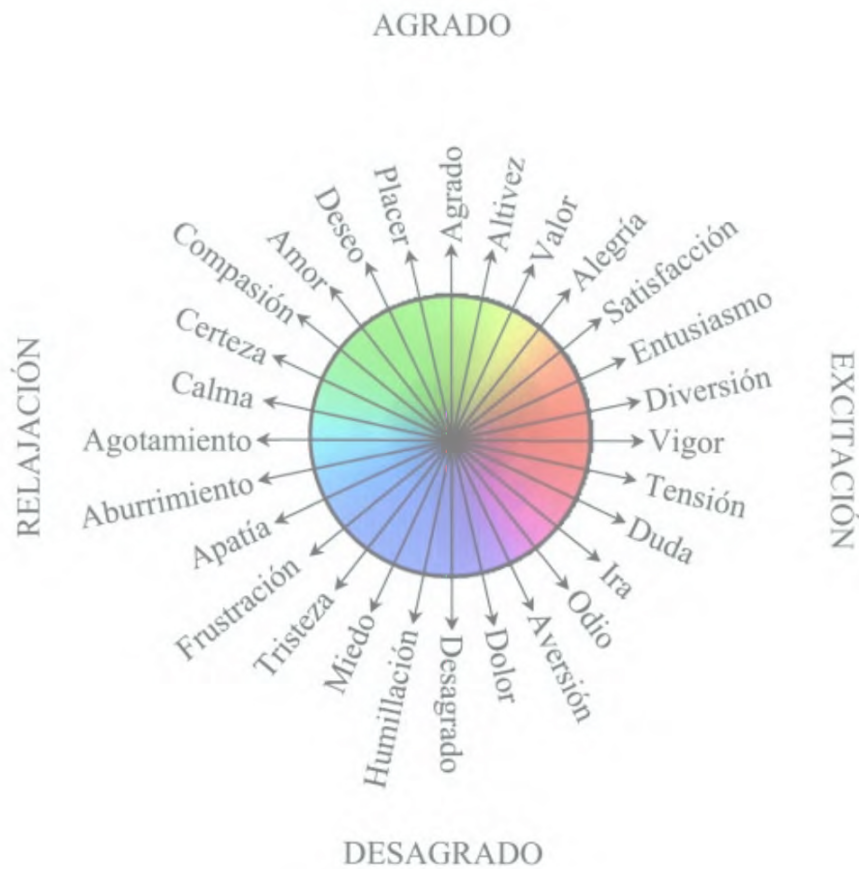


Figura 2.14: Modelo de Russell (Fuente: [www.redalyc.org](http://www.redalyc.org)).

### 2.2.3. Modelo de Lang

El modelo de Carl Lang, conocido como modelo de estados afectivos es uno de los modelos bidimensionales más utilizados. Se representa gráficamente, al igual que los anteriores, y permite modelar el estado afectivo de una persona. Representa los estados afectivos dentro de un plano que cuenta con dos ejes que permiten una mejor localización de los mismos, los ejes varían del nivel de Valencia (Positiva y Negativa) al nivel de Agitación (Alta y Baja). Esta representación gráfica esta compuesta por un plano con dos ejes en el que se pueden localizar los estados afectivos dependiendo de su relación dentro del modelo . Los ejes muestran el nivel de valencia, desde negativa a positiva, y el nivel de activación, desde baja hasta alta. Mediante esta distribución se facilita la percepción de los estados afectivos gracias a su representación gráfica [20] (Figura 2.15).

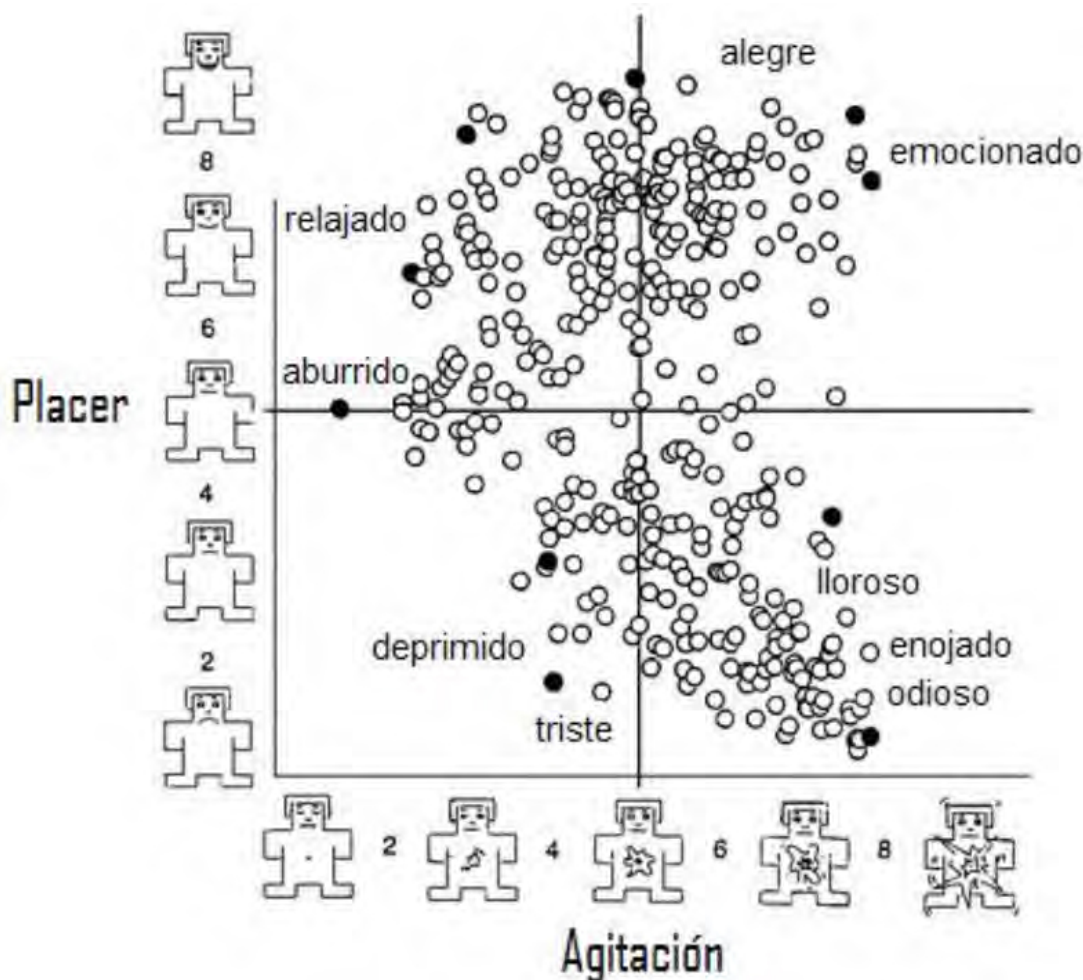


Figura 2.15: Lang (Fuente: [www.catarina.udlap.mx](http://www.catarina.udlap.mx)).

## 2.3. Detección de las emociones (ZenCar)

Las emociones originan pensamientos, reacciones fisiológicas y una respuesta. Para detectar estas emociones es necesario reconocer las respuestas que se producen. La conducción genera en una persona pensamientos asociados a emociones tanto positivas como negativas, además de los pensamientos que pueden existir en el conductor antes de entrar al coche y que condicionan las reacciones emocionales a la hora de conducir. Las respuestas fisiológicas como el sudor, aumento de la frecuencia cardíaca o la sequedad de boca se pueden percibir fácilmente y pueden ayudar a medir la intensidad de la emoción experimentada. Las respuestas provocadas por las emociones a la hora de conducir, como acelerar, pitar, hablar por teléfono, pegarse al vehículo de delante se podrían evitar si se pensasen fríamente, pero son acciones ejecutadas por el cerebro emocional.

Existe un proyecto llamado *ZenCar* especialmente diseñado para la medición y entrenamiento de las emociones durante la conducción. Dispone de equipos de medición que aúnan estas tres dimensiones de la emoción para estudiar la influencia de las emociones en la conducción. Una tecnología parecida a esta es utilizada por el ejército de Estados Unidos para el entrenamiento de sus pilotos o por deportistas de élite para prepararse psicológicamente ante grandes competiciones. En este caso, el proyecto *ZenCar* tiene como objetivo ayudar a alcanzar un estado de conducción *Zen* o *ZenDriving*, en el que las emociones no condicionan el comportamiento al volante del conductor. *ZenCar* emplea una serie de mecanismos para detectar las emociones del conductor, los cuales se muestran a continuación como un ejemplo de cómo monitorizar las acciones del usuario. Cada una de las reacciones fisiológicas y acciones que se detectan a continuación está relacionada de manera distinta con el comportamiento de los conductores, por lo que se muestra también una serie de gráficos sobre el impacto que tiene cada una de ellas sobre éstos [21].

### 2.3.1. Respuesta cardíaca

Se trata de estudiar el comportamiento del corazón del usuario durante la conducción. Un estado emocional intenso provoca que el corazón trabaje más y esto puede provocar un estado emocional alterado.

#### Frecuencia

La frecuencia cardíaca es fácil de medir y resulta extremadamente útil. Se basa en controlar las pulsaciones por minuto del corazón. Pueden ofrecer datos significativos sobre el estado emocional o físico del conductor.

## Pnn50

Es un indicador de la variación de la frecuencia cardíaca. Se utiliza para conocer la cantidad de estrés a la que está sometido el corazón. Se calcula teniendo en cuenta la distancia entre un latido y el siguiente y restándola a la del latido anterior. Se calcula el porcentaje de latidos en los que la diferencia es mayor de 50 milisegundos. Si este porcentaje es inferior al 3% significa que el usuario se encuentra en una situación de estrés excesiva.



Figura 2.16: Estudio cardíaco en conductores españoles (Fuente: [www.unespa.es](http://www.unespa.es)).

### 2.3.2. Ondas cerebrales

Se trata de los voltajes generados por las neuronas. Se pueden registrar y medir en función de tres índices.

#### Índice de Activación

Mide la excitación neuronal a consecuencia de pensamientos o estímulos. Permite conocer el nivel de relajación de la persona. La activación será baja si se encuentra relajada y será alta si está sometida a niveles altos de estrés. El nivel adecuado está entre 35 y 45 micro voltios.





Figura 2.17: Estudio de la activación en conductores españoles (Fuente: [www.unespa.es](http://www.unespa.es)).

## Índice de Atención

Mide la capacidad de percibir y procesar estímulos en momentos determinados, es decir, la cantidad de información que el cerebro puede captar. El nivel de atención depende de las variaciones en el nivel de activación; para conseguir un nivel de atención óptimo la activación debe estar en un nivel medio.

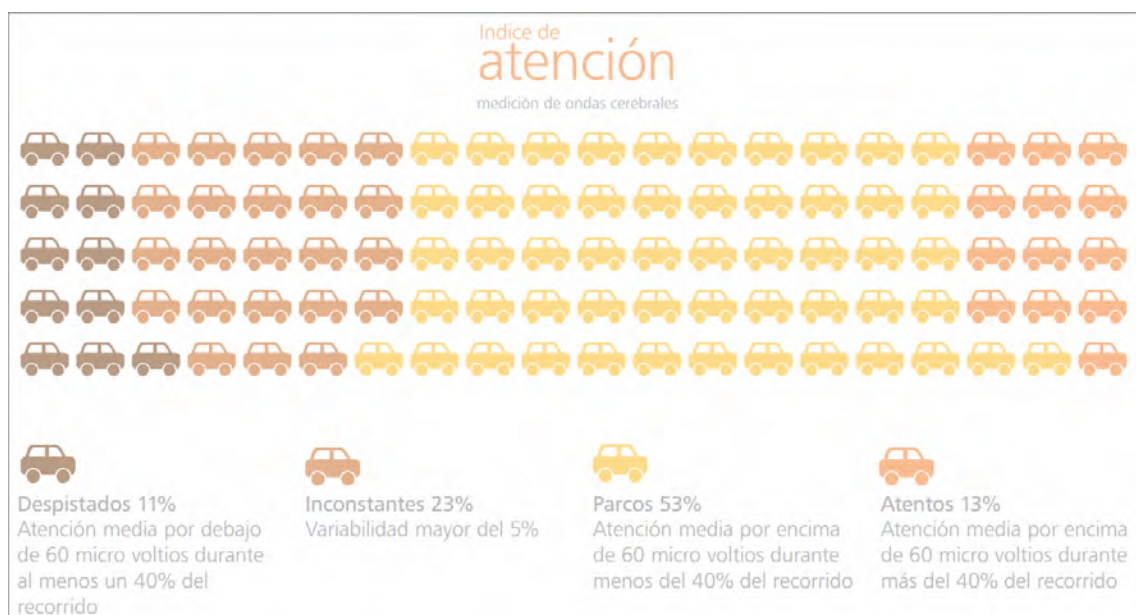


Figura 2.18: Estudio de la atención en conductores españoles (Fuente: [www.unespa.es](http://www.unespa.es)).

## Indicador Zen

Indica la capacidad de concentración en una única tarea independiente de sucesos internos y externos, los cuales se detectan mediante cámaras de vídeo que controlan el exterior del vehículo y las acciones y cambios en la expresión facial del conductor. De esta forma se conoce el estado emocional de la persona en lo referente a emociones positivas y negativas.

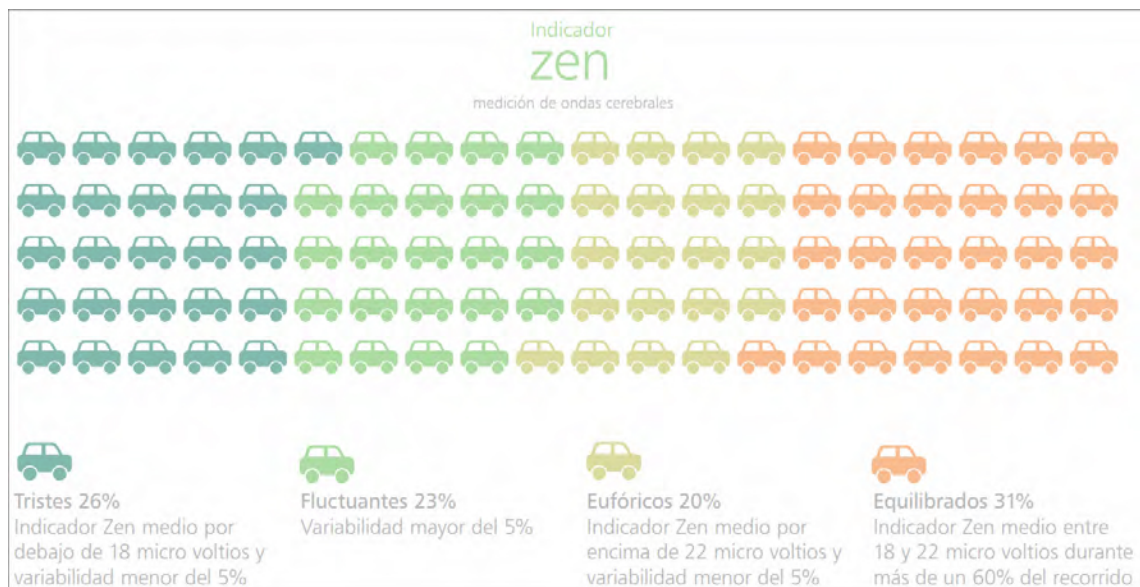


Figura 2.19: Estudio indicador Zen en conductores españoles (Fuente: [www.unespa.es](http://www.unespa.es)).

### 2.3.3. Parámetros del vehículo

El vehículo cuenta con aparatos de medida que registran los datos de conducción como aceleración, consumo, etc. Además, se utiliza tecnología de geolocalización para conocer la ubicación del vehículo en todo momento y así controlar los datos referentes a velocidad y recorrido.

Como resultado combinado de los cuatro estudios referenciados en las figuras 2.16, 2.17, 2.18 y 2.19 se obtienen tres tipos de conductores, de los cuales, el menos numeroso es el único que se puede considerar como seguro (Figura 2.20)



Figura 2.20: Estudio combinado en conductores españoles (Fuente: [www.unespa.es](http://www.unespa.es)).



# Capítulo 3

## Análisis del sistema

En este capítulo se expone una sola especificación de requisitos. Estos requisitos, tanto funcionales como no funcionales, ayudan a fijar la información básica sobre las necesidades que deberá satisfacer el sistema y fijan un objetivo final que se deberá tener en cuenta en la fase de diseño. Además, en este capítulo se expondrán también los casos de uso del sistema en forma de tablas y gráficamente.

### 3.1. Especificación de requisitos

La especificación de requisitos se realiza de forma tabular. Cada uno de los requisitos expuestos a continuación están contenidos en una tabla con el siguiente formato: (tabla 3.1):

Tabla 3.1: Modelo para requisitos.

Identificador	RX - YY		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción			

La descripción de los elementos que componen la tabla se expone a continuación:

- **Identificador.** Identifica cada uno de los requisitos unívocamente mediante un código. El formato que se utiliza es el siguiente:
  - **R.** Requisito.
  - **Y.** Toma el valor  $F$  si se trata de un requisito funcional y  $NF$  si se trata de un requisito no funcional.
  - **X.** Número que identifica cada requisito para distinguirlo del resto de su grupo.
- **Prioridad.** Indica la prioridad del cumplimiento de cada requisito para facilitar la gestión de objetivos durante el desarrollo. Existen tres niveles de prioridad: **alta**, **media** o **baja**.
- **Necesidad.** Indica la importancia del cumplimiento de cada requisito. Existen tres niveles de necesidad: **esencial** (si su cumplimiento es obligatorio), **deseable** (si su cumplimiento afecta en gran medida al proyecto, pero no es obligatorio) y **opcional** (si su cumplimiento no afecta al proyecto en gran medida).
- **Estabilidad.** Identifica la probabilidad de que un requisito cambia durante el desarrollo del proyecto. Existen tres niveles de estabilidad: **alta**, **media** o **baja**.
- **Descripción.** Descripción breve del requisito.

### 3.1.1. Requisitos funcionales

Un requisito funcional representa una funcionalidad del sistema que realiza el software del mismo. A continuación se exponen los requisitos funcionales del proyecto (tabla 3.2 hasta 3.17):

Tabla 3.2: RF - 01.

Identificador	RF - 01		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
<b>Descripción</b>	El sistema será capaz de detectar la posición de la cabeza del usuario para cada instante de tiempo definido.		

Tabla 3.3: RF - 02.

Identificador	RF - 02		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
<b>Descripción</b>	El sistema será capaz de detectar la posición de los ojos del usuario para cada instante de tiempo definido.		

Tabla 3.4: RF - 03.

Identificador	RF - 03		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema será capaz de detectar si los ojos del usuario están abiertos o cerrados con independencia el uno del otro para cada instante de tiempo definido.		

Tabla 3.5: RF - 04.

Identificador	RF - 04		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema será capaz de detectar si la boca del usuario está abierta o cerrada para cada instante de tiempo definido.		

Tabla 3.6: RF - 05.

Identificador	RF - 05		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema será capaz de detectar si el usuario está utilizando gafas para cada instante de tiempo definido.		

Tabla 3.7: RF - 06.

Identificador	RF - 06		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema será capaz de detectar si el usuario está sonriendo para cada instante de tiempo definido.		

Tabla 3.8: RF - 07.

Identificador	RF - 07		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema activará una alarma cuando considere que el usuario lleva demasiado tiempo con los ojos cerrados.		

Tabla 3.9: RF - 08.

Identificador	RF - 08		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema activará una alarma cuando considere que el usuario está bostezando.		

Tabla 3.10: RF - 09.

Identificador	RF - 09		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema activará una alarma cuando considere que el usuario está hablando.		

Tabla 3.11: RF - 10.

Identificador	RF - 10		
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema activará una alarma cuando considere que el usuario está bebiendo.		

Tabla 3.12: RF - 11.

Identificador	RF - 11		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema activará una alarma cuando considere que el usuario está distraído.		

Tabla 3.13: RF - 12.

Identificador	RF - 12		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema activará una alarma cuando considere que el usuario lleva demasiado tiempo sin mirar la carretera.		

Tabla 3.14: RF - 13.

Identificador	RF - 13		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema escribirá en un fichero toda la información obtenida para cada instante de tiempo definido.		

Tabla 3.15: RF - 14.

Identificador	RF - 14		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema será capaz de enviar la información obtenida al equipo donde se ejecuta el simulador de conducción.		

Tabla 3.16: RF - 15.

Identificador	RF - 15		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema deberá mostrar en pantalla la información obtenida en tiempo real.		

Tabla 3.17: RF - 16.

Identificador	RF - 16		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema contará con una interfaz gráfica para su uso.		

### 3.1.2. Requisitos no funcionales

Un requisito no funcional indica características sobre el funcionamiento del sistema, no funcionalidades en sí. A continuación se muestran los requisitos no funcionales del proyecto (tabla 3.2 hasta 3.17):

Tabla 3.18: RNF - 01.

Identificador	RNF - 01		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	La posición de la cabeza estará representada por tres valores numéricos (yaw, pitch, roll) en grados. Cada uno de ellos estará comprendido entre -180 y 180 grados.		

Tabla 3.19: RNF - 02.

Identificador	RNF - 02		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Descripción	La posición de los ojos estará representada por un <i>string</i> indicando la posición (derecha, centro, izquierda). No puede darse la superposición de varios valores al mismo tiempo.		

Tabla 3.20: RNF - 03.

Identificador	RNF - 03		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input checked="" type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Descripción	El sistema estará implementado en un proyecto WPF		

Tabla 3.21: RNF - 04.

Identificador	RNF - 04		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El usuario debe estar situado a una distancia de la cámara especificada para su correcto funcionamiento.		

Tabla 3.22: RNF - 05.

Identificador	RNF - 05		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input checked="" type="checkbox"/> Opcional
Estabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El <i>log</i> generado en cada ejecución del programa quedará guardado en un fichero <i>.txt</i> .		

Tabla 3.23: RNF - 06.

Identificador	RNF - 06		
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Estabilidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Descripción	El usuario podrá alternar entre el modo de video <i>infrarrojo</i> y <i>color</i> para activar distintas funcionalidades de la cámara.		

## 3.2. Casos de uso

En esta sección se muestran los casos de uso del sistema. Los casos de uso representan los pasos a seguir para realizar los diferentes procesos del sistema, una secuencia de interacciones entre el sistema y un actor (usuario) en respuesta a determinados eventos. Primero se muestran los casos de uso en forma de tabla y más adelante se representarán de forma gráfica.

### 3.2.1. Descripción tabular de casos de uso

El formato de tabla que se utiliza para representar los casos de uso es el siguiente:

Tabla 3.24: Modelo para casos de uso.

Identificador	CU - XX
Título	
Objetivo	
Precondiciones	
Postcondiciones	
Escenario de éxito	

La descripción de los elementos que componen la tabla se expone a continuación:

- **Identificador.** Identifica cada uno de los casos de uso unívocamente mediante un código. El formato que se utiliza es el siguiente:
  - **CU.** Caso de uso.
  - **XX.** Número que identifica cada caso de uso.
- **Título.** Nombre con el que se identifica el caso de uso.
- **Objetivo.** Meta a la que se quiere llegar con el caso de uso definido.
- **Precondiciones.** Serie de condiciones necesarias para la consecución del caso de uso.
- **Postcondiciones.** Condiciones en las que se encuentra el sistema tras la consecución del caso de uso.
- **Escenario de éxito.** Sucesión de eventos que llevan a la consecución del caso de uso.

A continuación se exponen los casos de uso del proyecto (tablas 3.2) hasta 3.17):



Tabla 3.25: CU - 01.

Identificador	CU - 01
<b>Título</b>	Conexión con el servidor
<b>Objetivo</b>	Conectar con el equipo en el que se ejecuta el simulador de conducción para enviar la información recogida.
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>■ El programa debe comenzar a ejecutarse.</li> <li>■ En la interfaz debe mostrarse lo que la cámara está captando.</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>■ La conexión está establecida y los mensajes pueden empezar a enviarse al simulador.</li> </ul>
<b>Escenario de éxito</b>	<ul style="list-style-type: none"> <li>■ Ejecutar la aplicación en el equipo que actúa como cliente.</li> <li>■ Ejecutar el simulador de conducción en el equipo que actúa como servidor.</li> <li>■ Establecer conexión entre los dos equipos.</li> <li>■ La información comienza a enviarse en cuanto esté disponible.</li> </ul>

Tabla 3.26: CU - 02.

Identificador	CU - 02
<b>Título</b>	Reconocimiento del esqueleto por el dispositivo Kinect v2.0.
<b>Objetivo</b>	Ser reconocido por el dispositivo Kinect v2.0 para comenzar a obtener información.
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>■ La interfaz de usuario debe mostrar las imágenes que la cámara está captando.</li> <li>■ El usuario debe estar situado a una distancia adecuada del dispositivo para poder detectar su cuerpo.</li> <li>■ Para facilitar el proceso de reconocimiento, el usuario debe levantar ambos brazos a sus lados.</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>■ El esqueleto del usuario es reconocido por el dispositivo.</li> </ul>
<b>Escenario de éxito</b>	<ul style="list-style-type: none"> <li>■ Ejecutar la aplicación.</li> <li>■ Colocarse a una distancia adecuada.</li> <li>■ Levantar los brazos para facilitar el reconocimiento.</li> <li>■ Ser reconocido por el dispositivo.</li> </ul>

Tabla 3.27: CU - 03.

Identificador	CU - 03
Título	Reconocimiento facial.
Objetivo	Reconocimiento de la cara del usuario por el dispositivo Kinect v2.0.
Precondiciones	<ul style="list-style-type: none"> <li>■ El esqueleto del usuario ha sido reconocido.</li> </ul>
Postcondiciones	<ul style="list-style-type: none"> <li>■ La cara del usuario es reconocida por el dispositivo.</li> <li>■ La interfaz muestra una señalización de los puntos principales de la cara (nariz, ojos, boca).</li> <li>■ La interfaz muestra la información obtenida en cada <i>frame</i> por el dispositivo.</li> </ul>
Escenario de éxito	<ul style="list-style-type: none"> <li>■ Una vez ha sido detectado el esqueleto, la cara es detectada automáticamente por la aplicación.</li> <li>■ La cara del usuario es señalizada en la interfaz y la información obtenida y las alarmas comienzan a mostrarse en pantalla.</li> </ul>

Tabla 3.28: CU - 04.

Identificador	CU - 04
Título	Gestión de la información.
Objetivo	Gestión de la información obtenida por el dispositivo. La información se muestra en pantalla, se escribe en logs que quedan guardados en el quipo y es enviada al equipo que actúa como servidor.
Precondiciones	<ul style="list-style-type: none"> <li>■ La cara del usuario ha sido detectada por el dispositivo.</li> <li>■ En la interfaz se muestra en el mensaje “Escribiendo”.</li> </ul>
Postcondiciones	<ul style="list-style-type: none"> <li>■ La información obtenida en cada <i>frame</i> se muestra en pantalla.</li> <li>■ La información obtenida en cada <i>frame</i> se escribe en un fichero <i>.txt</i>.</li> <li>■ La información obtenida se envía al equipo que actúa como servidor.</li> </ul>
Escenario de éxito	<ul style="list-style-type: none"> <li>■ Una vez la cara ha sido detectada por el dispositivo, la información comienza a gestionarse.</li> <li>■ El usuario puede comenzar a conducir.</li> </ul>

Tabla 3.29: CU - 05.

Identificador	CU - 05
Título	Reconocimiento facial.
Objetivo	Reconocimiento de la cara del usuario por el dispositivo Kinect v2.0.
Precondiciones	<ul style="list-style-type: none"> <li>■ El esqueleto del usuario ha sido reconocido.</li> </ul>
Postcondiciones	<ul style="list-style-type: none"> <li>■ La cara del usuario es reconocida por el dispositivo.</li> <li>■ La interfaz muestra una señalización de los puntos principales de la cara (nariz, ojos, boca).</li> <li>■ La interfaz muestra la información obtenida en cada <i>frame</i> por el dispositivo.</li> </ul>
Escenario de éxito	<ul style="list-style-type: none"> <li>■ Una vez ha sido detectado el esqueleto, la cara es detectada automáticamente por la aplicación.</li> <li>■ La cara del usuario es señalizada en la interfaz y la información obtenida y las alarmas comienzan a mostrarse en pantalla.</li> </ul>

## Capítulo 4

# Arquitectura y diseño del sistema

En este capítulo se expone el diseño del sistema y se definen los distintos componentes por los que esta formado, así como la interacción entre ellos.

En este sistema podemos encontrar cinco módulos principales que interactúan entre sí: la interfaz gráfica, la gestión de *frames*, el seguimiento ocular, el sistema de alarmas y la conexión mediante sockets. Cada uno de estos componentes representa una de las funcionalidades principales del sistema. En la figura 4.1 se puede ver un diagrama que describe la relación entre estos módulos.

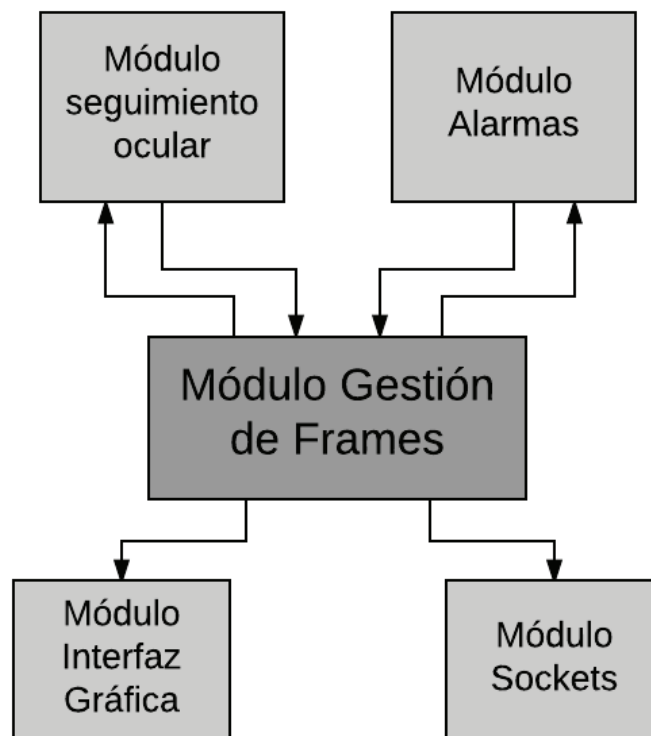


Figura 4.1: Diagrama de módulos de la arquitectura del sistema.

#### 4.0.1. Módulo interfaz gráfica

Esta parte del sistema está diseñado en XAML. El módulo permite representar toda la información que maneja el sistema por pantalla, permitiendo al usuario ver el estado de la monitorización y el resto de actividades en tiempo real. Además permite al usuario interactuar con el sistema para activar o desactivar diferentes funcionalidades.

Todos los cambios que se realizan en el módulo de gestión de frames actualizan las variables de éste módulo y en la pantalla se muestran los datos actualizados.

#### 4.0.2. Módulo gestión de *frames*

Este módulo se encarga de la gestión principal del funcionamiento del sistema. El sistema utiliza diferentes tipos de imágenes captadas por los dos tipos de cámara que utiliza el dispositivo Kinect: cámara infrarroja y cámara de alta definición. Los fotogramas de ambos flujos de imágenes se captan a la vez, de modo que hay que gestionar cada fotograma de manera distinta dependiendo de las operaciones que se estén realizando. Cada fotograma tiene distintas características y lleva consigo una gran cantidad de información que debe ser procesada por el sistema. Este módulo se encarga de gestionar los *frames* y la información que contienen para que el sistema se ejecute de forma fluida.

Dentro de este módulo se gestiona también la información procedente de los módulos de seguimiento ocular y de alarmas. Se recibe esta información y se junta con la información obtenida en el propio módulo, es decir, la obtenida por el dispositivo Kinect. Toda esta información sigue diversos caminos. Por una parte se escribe en ficheros de texto que quedan guardados en el equipo, también es utilizada por el módulo interfaz para mostrarlo por pantalla y, finalmente, también es recogida por el módulo sockets, que será el encargado de enviar la información a otro equipo. El módulo de alarmas utilizará esta información para actualizarse y el módulo de seguimiento ocular utilizará los nuevos *frames* gestionados para actualizar la posición del ojo.

#### 4.0.3. Módulo seguimiento ocular

Este módulo contiene una funcionalidad adicional que no puede ser ejecutado por el dispositivo Kinect, se trata de la detección de las pupilas. Dado que la cámara no es capaz de proveer esta información, es decir, no está presente en la información contenida en los fotogramas, es necesario un módulo independiente que ejecute un método con dicha funcionalidad.

Este módulo utiliza los *frames* de alta definición obtenidos del módulo de gestión de *frames* y detecta la posición de la pupila del usuario. Cuando se ha determinado la posición, envía esta información de vuelta al módulo de gestión de frames.

#### **4.0.4. Módulo Alarmas**

Este módulo gestiona e interpreta toda la información recogida por el módulo de gestión de frames y genera nueva información basada en la anterior que puede resultar de ayuda. Esta nueva información se envía de vuelta al módulo de gestión de frames.

#### **4.0.5. Módulo sockets**

Para finalizar, toda la información recogida por el módulo de gestión de frames debe ser enviada al equipo donde se ejecuta el simulador de conducción. Este módulo se encarga de establecer una conexión entre los dos equipos y gestionar la comunicación entre ellos, enviando la información.

# Capítulo 5

## Implementación del sistema

En este capítulo se muestra la implementación del sistema, exponiendo y justificando las decisiones tomadas. Como ya se ha mencionado, la implementación del sistema persigue dos objetivos diferenciados, por un lado la monitorización del usuario y por otro, la comunicación con el simulador mediante sockets. Por lo tanto, en este capítulo se hablará inicialmente de cómo se ha implementado el sistema en general para luego centrarse en lo relativo a los dos objetivos.

Inicialmente se evalúan cuáles son las características que se van a monitorizar para así conocer qué funcionalidades es necesario implementar. Como ya se ha indicado en el estado del arte, el dispositivo Kinect v2.0 cuenta con varias funcionalidades cada una con distintas características.

Se han distinguido tres fuentes de información de las que obtener los datos en el sistema: monitorización facial utilizando los métodos ofrecidos por el dispositivo, monitorización de los ojos y sistema de alarmas. Las dos primeras fuentes de información son las que requieren la monitorización del usuario, pero no utilizando los mismos medios.

Para la monitorización facial dentro de vehículos existe un inconveniente que supone un gran problema para la detección por vídeo, este problema es la iluminación. Cuando un vehículo se encuentra circulando, las condiciones lumínicas pueden variar constantemente y no ser las adecuadas para detectar los rasgos del conductor, ya sea por la presencia de túneles o, peor aún, por conducir por la noche. Por lo tanto, para obtener esta información se utilizará la cámara de infrarrojos integrada en el dispositivo Kinect, la cual funciona independientemente de la iluminación.

Para la monitorización de los ojos del usuario se podría utilizar también la cámara infrarroja, pero debido al método utilizado es necesario utilizar la cámara RGB integrada en el dispositivo, la cual tiene una resolución mucho mayor y, como su propio nombre indica, diferencia los colores. Más adelante se explicarán los motivos con más detalle.

La tercera fuente de información es resultado de la interpretación de los datos obtenidos en las dos anteriores, por lo que no necesita ningún sistema de monitorización. Partiendo de esta base, se pasa a explicar el funcionamiento del sistema.

## 5.1. Funcionamiento del sistema

El programa desarrollado se ha implementado en lenguaje C, ya que este es uno de los lenguajes en los que Microsoft ofrece su SDK para Kinect, por lo tanto se cuenta con una mayor cantidad de recursos además de los ejemplos de funcionalidades ya existentes mencionados en el estado del arte. El sistema está implementado en un proyecto WPF y su interfaz gráfica está desarrollada en XAML. Se utiliza como modelo el proyecto creado por el departamento de Automática para el trabajo fin de máster *Detección del estado emocional de un conductor mediante el uso de imágenes infrarrojas* [33]. El funcionamiento del programa se divide en varias etapas que se presentan a continuación.

Al ejecutar el programa, se accede al dispositivo y a los flujos de información que se vayan a utilizar, en este caso las imágenes en infrarrojo, las imágenes a color, la detección de esqueletos y el reconocimiento facial. Entonces se comienza a mostrar en pantalla las imágenes captadas por la cámara, en este caso en modo infrarrojo o IR (se ha puesto esta opción por defecto), pudiendo cambiar en cualquier momento al modo en color. Una vez se puedan observar la imágenes en pantalla comienza la fase de **detección de esqueletos**.

Se ha observado que el dispositivo reconoce mejor los cuerpos humanos si éstos no se encuentran demasiado cerca del objetivo, por lo que la cámara debe colocarse a una distancia adecuada. El dispositivo puede reconocer hasta seis esqueletos a la vez, pero en este trabajo sólo es necesario tener en cuenta a una persona. Si el detector de esqueletos reconoce alguno, se continua con la siguiente fase, la **detección de la cara**.

La detección facial actúa en paralelo a la detección de esqueletos, pero no puede funcionar sin conocer la posición del cuerpo de la persona, por lo que, una vez que se ha detectado el esqueleto, la detección facial comienza. En caso de que aún se desconozca esta información se continúa buscando el esqueleto.

Cuando ya se ha encontrado la cara del usuario, se analizan en ella las características que nos permite detectar el SDK de Microsoft y comienza la monitorización. En el siguiente diagrama se muestra de forma gráfica el proceso de detección para facilitar su comprensión:

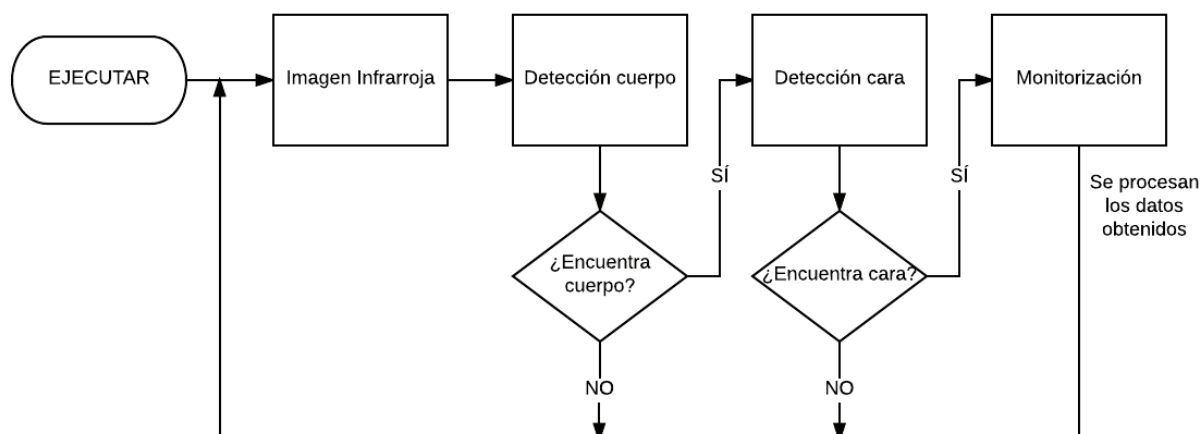


Figura 5.1: Diagrama de flujo de detección de la cara del usuario.



A continuación se profundiza un poco más en las fases de detección mencionadas en los párrafos anteriores, y posteriormente se explica la monitorización.

### 5.1.1. Flujo de imágenes

Al iniciar la aplicación se activa la cámara y empiezan a recibirse los distintos flujos de imágenes. En este caso utilizaremos las imágenes infrarrojas para detectar los rasgos de la cara y las imágenes a color para la posición de las pupilas. La cámara infrarroja captura imágenes con una resolución de 512 x 424 píxeles y su frecuencia es de 30 Hz (por cada segundo que pasa se han capturado 30 imágenes nuevas), mientras que la cámara a color cuenta con una resolución de 1920 x 1080 píxeles. Cuando una de las imágenes o *frames* es recibido, éste se recoge en la sección XAML del programa y se muestra por pantalla a través de la interfaz tal y como se muestra en las figuras 5.2 y 5.3 .

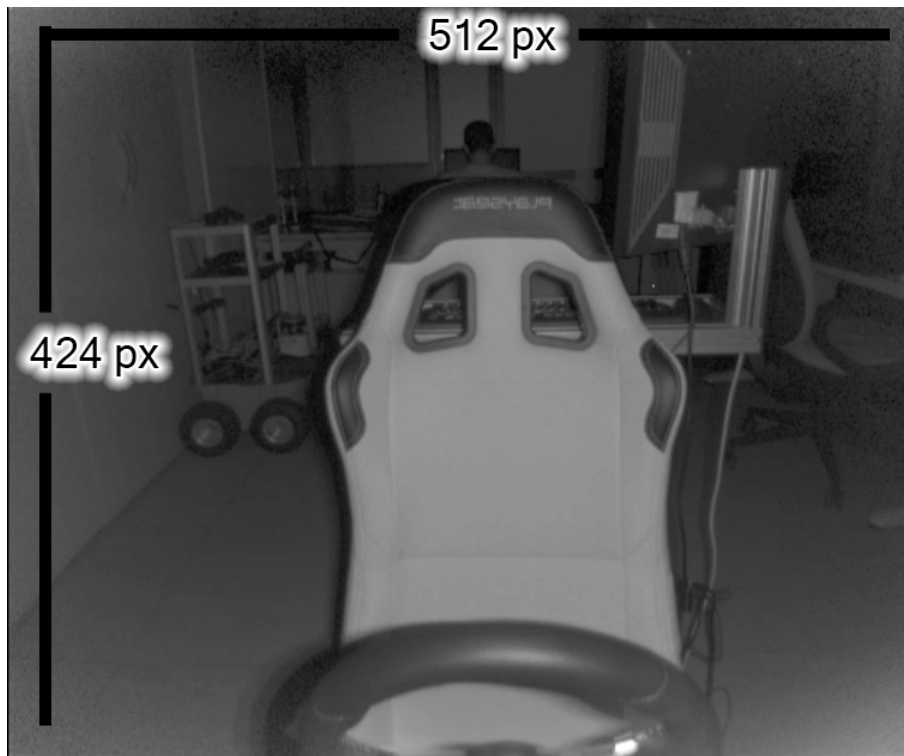


Figura 5.2: Imagen tomada por la cámara infrarroja de la posición a ocupar por el conductor en el simulador.



Figura 5.3: Imagen tomada por la cámara de alta definición de la posición a ocupar por el conductor en el simulador.

Ambos flujos de imágenes se capturan simultáneamente, pero no es necesario mostrar los dos por pantalla mismo tiempo, además de las ralentizaciones que esto supone en el rendimiento del programa. Por este motivo se ha diseñado un sistema de “modos” para alternar entre ambos flujos desde la interfaz gráfica, el cual se puede ver en la figura 5.4 . Al entrar en el *modo color* se activan las funcionalidades nuevas que se han añadido para la detección de las pupilas, de las cuales se hablará más adelante.

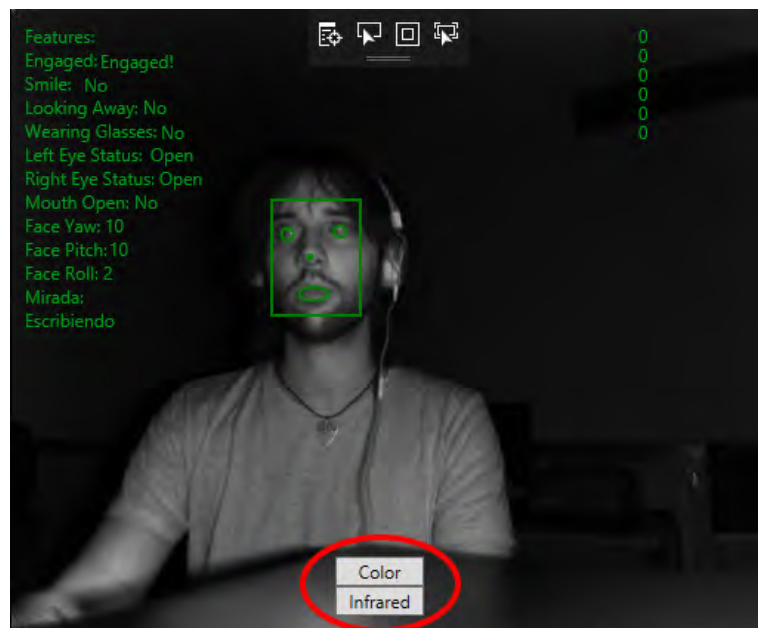


Figura 5.4: Botones de selección de modo de imagen (señalados en un círculo rojo).

### 5.1.2. Detección de esqueletos

El proceso de encontrar el la posición del cuerpo se divide en dos fases: crear un mapa de profundidad utilizando iluminación estructurada y deducir la posición del cuerpo utilizando aprendizaje automático, todo ello realizado por el dispositivo Kinect.

Para crear el mapa de profundidad se analiza un patrón de manchas de luz infrarroja. Esta técnica de analizar un patrón conocido se denomina *iluminación estructurada*, y se basa en proyectar un patrón conocido sobre una escena y deducir la profundidad de las deformaciones originadas en el patrón. La cámara combina esta técnica con otras dos: profundidad por enfoque y profundidad por estéreo. La primera se obtiene partiendo de que los objetos situados más lejos de la cámara se verán más borrosos. La segunda se basa en que si observas un objeto desde otro ángulo, cuanto más cerca esté, mayor será el giro que se observa al desplazarse. Por eso mismo el dispositivo proyecta desde una lente y observa desde otra (Figura 5.5).



Figura 5.5: Cámara Kinect v1 (Fuente: [www.users.dickinson.edu](http://www.users.dickinson.edu)).

En la segunda fase, las partes del cuerpo se detectan a partir de una base de datos de esqueletos utilizando un árbol de decisión randomizado. Este proceso es más complejo, por lo que, en caso de necesitar una descripción más detallada, se puede acudir a la bibliografía [32] (Figura 5.6).



Figura 5.6: Base de datos de esqueletos utilizada por *Kinect* para detectar al usuario.

### **5.1.3. Detección facial**

Como ya se ha explicado, este proceso se ejecuta de forma paralela a la detección de esqueletos. En el momento en que el cuerpo del usuario es detectado, el detector de caras puede funcionar. Este detector también cuenta con una base de datos de caras, la cual se utiliza para realizar un Análisis de Componentes Principales (PCA) y comparárlas con las imágenes capturadas por la cámara. Llegados a este punto, se puede comenzar a explicar con detalle la monitorización del usuario en los apartados siguientes.

### **5.1.4. Monitorización del usuario**

A continuación se explica en detalle el funcionamiento de cada uno de los aspectos que se van a monitorizar en el conductor.

#### **Monitorización facial**

Tal y como se ha comentado anteriormente, aunque la simulación de conducción se realiza en un entorno cerrado y con condiciones lumínicas controladas, hay que tener en cuenta las condiciones que se aplicarían en un entorno urbano real. Por lo tanto, en esta fase se utilizará la cámara infrarroja que ofrece Kinect, de tal forma que la iluminación no suponga un problema a la hora realizar la monitorización.

Las cámaras infrarrojas pueden percibir la radiación infrarroja emitida por distintos cuerpos, la cual depende de la temperatura de éstos, y muestran una imagen monocromática en la que se representa esta temperatura. Esto puede causar problemas si se utiliza de día por la radiación emitida por el sol, por lo que la cámara con la que se trabaja no utiliza el mismo sistema. Esta cámara emite un haz infrarrojo y es capaz de detectar la reflexión de este haz sobre los cuerpos, de esta forma la radiación solar no interfiere en el proceso (Figuras 5.7 y 5.8).



Figura 5.7: Imagen del conductor captada por la cámara infrarroja.

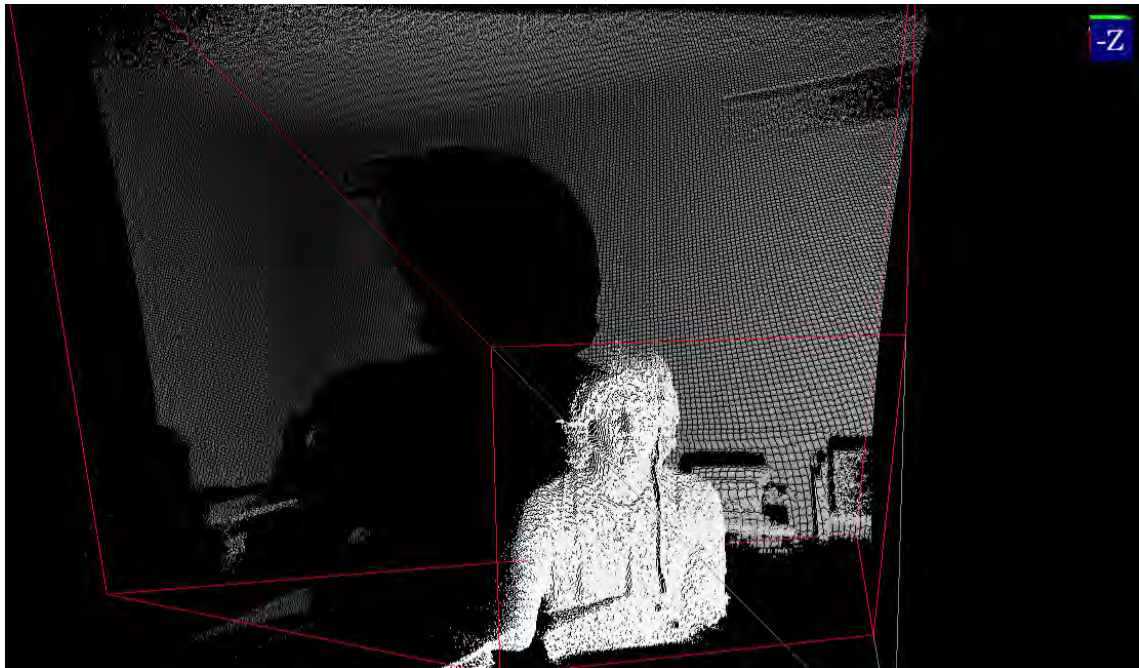


Figura 5.8: Haz de luz infrarroja que proyecta la cámara.

Gracias al SDK de Microsoft, en cada fotograma captado por la cámara infrarroja podemos detectar las siguientes características:

- ***BoundingBoxInInfraredSpace***: Un rectángulo que se muestra en la interfaz delimitando la zona de la cara.
- ***PointsInInfraredSpace***: Los puntos de la cara detectados por la cámara que se pueden utilizar, tales como ojo derecho, ojo izquierdo, nariz, etc.
- ***FaceEngagement***:
- ***LeftEyeClosed***: Indica si el ojo izquierdo está cerrado.
- ***RightEyeClosed***: Indica si el ojo derecho está cerrado.
- ***LookingAway***: Indica cuando el usuario no se encuentra mirando al frente, basándose únicamente en la posición de la cabeza.
- ***Glasses***: Indica si el usuario está usando gafas.
- ***MouthOpen***: Indica si la boca del usuario está abierta.
- ***MouthMoved***: Indica si los puntos asignados a la boca se han desplazado entre sí.
- ***Happy***: Indica si el usuario está sonriendo.
- ***RotationOrientation***: Valores que indican la posición de la cabeza.

Utilizando esta información se puede mostrar en la interfaz gráfica los puntos de interés sobre la cara del usuario para conocer su posición en todo momento. Además se muestra también el estado de las variables que han sido descritas en el párrafo anterior, tal y como se muestra en la figura 5.9.

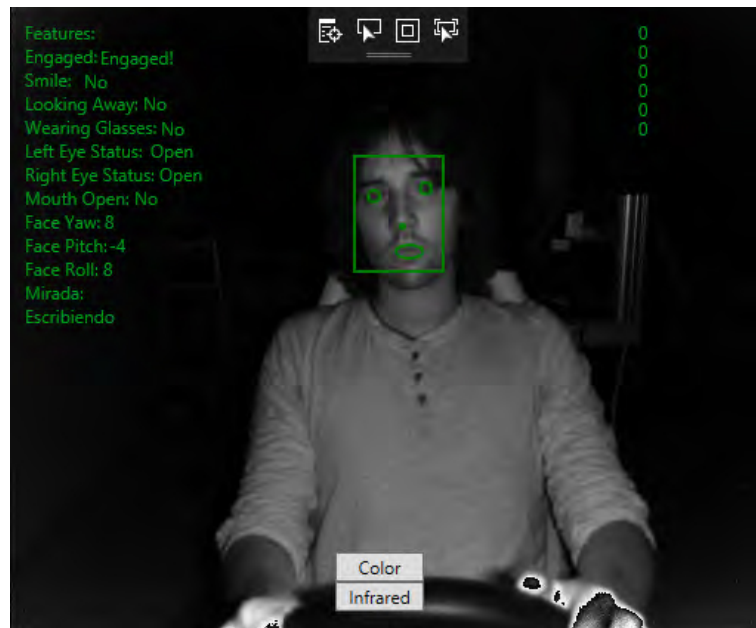


Figura 5.9: Ejemplo de la cara del conductor detectada por la cámara infrarroja.



Aunque, para mostrar los resultados que aparecen en la figura 5.9 es necesario realizar ciertas modificaciones a los valores que representan la posición de la cabeza. Esta posición viene representada con cuatro valores, pero para facilitar su interpretación se han transformado en tres ángulos conocidos como *roll*, *pitch* y *yaw* que representan la posición de la cabeza con un rango de  $-180^\circ$  a  $180^\circ$  para cada uno de los movimientos que pueden observarse en la figura 5.10.

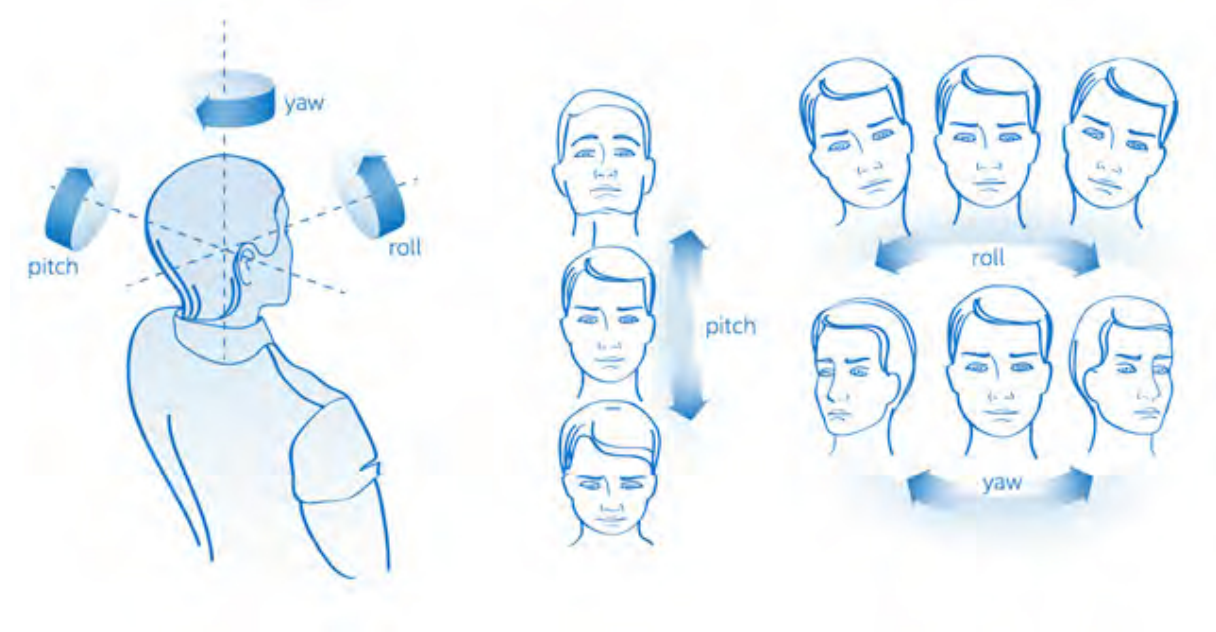


Figura 5.10: Posiciones de la cabeza (Fuente: 65.media.tumblr.com).

## Monitorización ocular

El siguiente aspecto que se va a monitorizar en el usuario es la posición de las pupilas. La cámara ofrece muchas características que suponen una cantidad de información considerable, pero se considera que conocer la posición de los ojos es esencial para garantizar la seguridad del conductor. Sin esta información el usuario puede tener la cabeza perfectamente alineada con la carretera y el sistema consideraría que no ocurre nada malo, mientras que el usuario puede estar mirando en otra dirección, creando una situación de peligro.

Para conseguir implementar esta nueva funcionalidad se han utilizado las librerías de procesamiento de imagen *EmguCV*. *EmguCV* es una versión de la librería *OpenCV* que permite utilizar las funciones que ofrece ésta desde lenguajes compatibles con .NET como C#. Utilizando esta librería se pueden realizar ciertas operaciones y modificaciones sobre las imágenes captadas por la cámara Kinect.

Para determinar la posición de la pupila, primero hay que detectarla, lo cual puede realizarse de diversas formas. Se puede detectar buscando las formas redondas que aparezcan en la imagen o diferenciando las distintas partes del ojo por el grado de oscuridad que presenta la pupila con respecto al resto del ojo. El primer método se puede llevar a cabo mediante una función que ofrece la librería *EmguCV*, pero la parte de la imagen correspondiente al ojo es sólo una pequeña parte de la imagen global, por lo que las imágenes aisladas de los ojos son demasiado pequeñas y, por lo tanto, están demasiado pixeladas como para poder diferenciar una forma circular. Por este motivo se utilizará el segundo método descrito.

Mediante este método se diferencia entre las partes más claras del ojo del usuario y las partes más oscuras, como la pupila y el iris. Por lo tanto, se ha considerado como necesario utilizar la cámara de alta definición integrada en el dispositivo Kinect. Primero, por el motivo mencionado anteriormente, el ojo del usuario se corresponde con una pequeña parte de la imagen y es necesario contar con imágenes con la mayor resolución posible, en este caso 1920x1080 píxeles. La segunda razón es la distinción de los colores. La cámara infrarroja capta una imagen monocromática en escala de grises en la que las diferencias entre las partes claras y oscuras del ojo no son tan destacadas, al menos no a la distancia a la que se coloca la cámara del usuario (distancia necesaria para el funcionamiento adecuado del dispositivo). Por lo tanto se utilizarán las imágenes captadas por la cámara a color de alta resolución (como la que se puede observar en la figura 5.11) y se someterán a un proceso que se explica más adelante.



Figura 5.11: Imagen del conductor captada por la cámara de alta definición.

Antes del procesamiento de imágenes es necesario conocer una de las características que nos permite utilizar la cámara de alta definición y que será necesaria más adelante. La cámara a color, al igual que la cámara infrarroja, reconoce una serie de puntos pertenecientes al rostro humano, pero, en este caso, al contar con una mayor resolución, se tiene acceso a muchos más puntos, los cuales se pueden observar en la figura 5.12 en color azul.





Figura 5.12: Puntos de la cara detectados por la cámara de alta definición.

Todos estos puntos pueden ser utilizados para monitorizar expresiones más complejas y poder detectar ciertas emociones en el conductor, pero, para el presente trabajo, sólo se hace uso de los puntos pertenecientes a los ojos. La importancia de estos puntos es muy alta, ya que el primer paso para el procesado de la imagen consiste en “recortar” una región de interés dentro de la imagen, en este caso los ojos. El problema reside en que esta región de interés no siempre tiene las mismas dimensiones, varía con la distancia a la cámara y las dimensiones del ojo y la cabeza del usuario. Por eso es necesario utilizar estos puntos que se adaptan al usuario en todo momento. Por ejemplo, si el usuario se encuentra más alejado de la cámara, la región correspondiente a los ojos será una porción de la imagen más pequeña que si el usuario estuviera posicionado más cerca. A continuación se explica el procesado al que se someten las imágenes.

Para detectar la posición de la pupila en una imagen captada por la cámara, utilizando las herramientas proporcionadas por la librería *EmguCV*, se somete dicha imagen al siguiente proceso:

- **Seleccionar Region de interes (ROI):** La imagen inicial con la que se comienza el proceso contiene todo lo que la cámara captura, mientras que, para esta fase, sólo se necesita la zona del ojo. Por lo tanto, se marca como zona de interés un rectángulo delimitando la zona del ojo que se utilizará para trabajar, ayudándose de los puntos de alta definición de los que se ha hablado anteriormente (Figura 5.13).

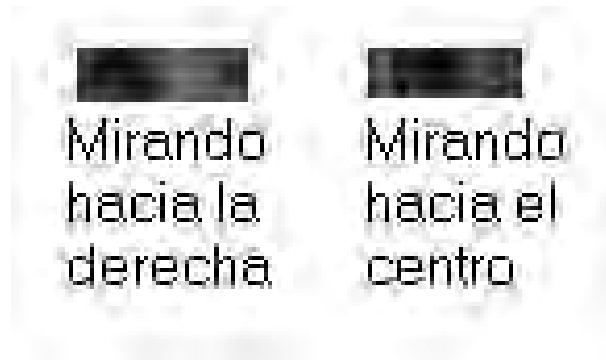


Figura 5.13: Región de interés en una imagen captada por la cámara de alta definición.

Sólo es necesario monitorizar uno de los ojos del usuario, ya que el otro ojo se comportará de manera similar. El problema surge cuando el usuario gira la cabeza, de forma que el ojo que se está monitorizando puede obstaculizar la visión del ojo seleccionado (como se ve en la figura 5.14). Para solucionar esto, el programa delimitará la zona de interés en el ojo izquierdo por defecto, salvo que el valor del parámetro *YAW* expresado en grados sea negativo. En ese caso se utilizará el ojo derecho, que permanece visible.

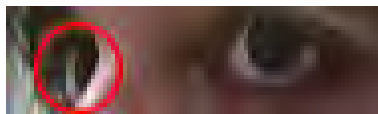


Figura 5.14: Ojo obstaculizado por la posición de la cabeza (señalado por círculo rojo).

Otro problema que se ha encontrado durante la implementación es el posicionamiento de los puntos de alta definición dependiendo del usuario. Los puntos que señalan los ojos no siempre se colocan perfectamente rodeando el ojo en todas las personas, de modo que la región de interés puede seleccionarse en una zona en la que no está el ojo. Esto supone un gran problema, ya que se necesitan estos puntos para controlar las dimensiones del rectángulo de la región de interés, que varían con la distancia entre el usuario y la cámara.

La solución a la que se ha llegado se basa en utilizar los puntos de alta definición y los puntos básicos que capta la cámara infrarroja de manera conjunta. Entre los puntos básicos a los que se tiene acceso existe uno para cada ojo, que representa la posición central de éste, por lo tanto, se puede utilizar como método fiable para encontrar la posición del ojo (se ha probado con distintas personas y no ocurre el error observado en los otros puntos). Los puntos de alta definición se seguirán usando para calcular las dimensiones de la región de interés, ya que, aunque su posición sea errónea a veces, las dimensiones del ojo delimitado por puntos siguen siendo proporcionales a las del ojo real independientemente de la distancia. En la figura 5.15 se pueden reconocer los puntos que marcan el centro del ojo en color verde y los que determinan las dimensiones en azul.

Los ajustes que se realizan para obtener la dimensión exacta de la región de interés se calculan mediante porcentajes y no valores dicretos, debido a la variabilidad que produce la distancia a la cámara. De esta manera, tanto en distancias cortas como largas, se pueden obtener regiones de interés proporcionales a la dimensión real del ojo del usuario.



Figura 5.15: Puntos de la cara detectados por la cámara infrarroja y de alta definición.

- **Convertir imagen a escala de grises:** Como la única información que se necesita es la diferencia entre la parte blanca del ojo y las partes más oscuras como el iris y la pupila, se transforma la imagen a color en una imagen en escala de grises (Figura 5.16). Así se puede realizar la comparación basándose en el nivel de negro.



Mirando  
hacia la  
derecha



Mirando  
hacia el  
centro

Figura 5.16: Región de interés en escala de grises.

- **Filtrar imagen:** A continuación se somete la imagen a un filtro mediante el cual todos los puntos de la imagen con una tonalidad de negro por encima de cierto umbral se convierten al color blanco, el resto se convierten a negro (Figura 5.17). De esta forma se obtiene una imagen en la que las partes blancas se corresponden con las zonas más oscuras, las cuales deberían corresponderse con la pupila.



Mirando  
hacia la  
derecha



Mirando  
hacia el  
centro

Figura 5.17: Región de interés tras someterla a un filtro de blanco y negro.

- **Filtro de erosión:** Tras el filtro anterior, no todas las partes blancas son parte de la pupila, lo que puede ocasionar un mal funcionamiento del método, por lo que se somete la imagen anterior a un filtro de erosión, mediante el que se pueden eliminar los puntos que forman los bordes de las partes blancas. Este filtro se aplica con un número determinado de iteraciones, por lo que, al aplicarse varias veces sobre las zonas blancas más pequeñas (originadas por partes como las pestañas), éstas desaparecen. Así sólo permanecen en la imagen puntos en blanco pertenecientes a las partes blancas más grandes, que corresponden a la pupila y el iris. El resultado tras aplicar el filtro de erosión puede observarse en la figura 5.18.

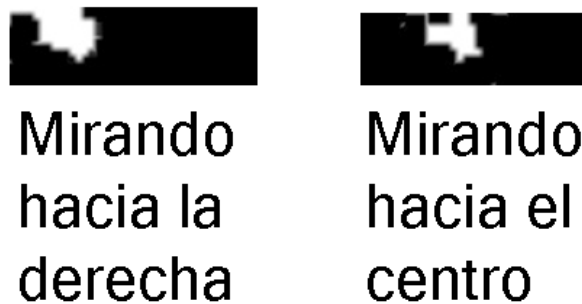


Figura 5.18: Región de interés tras someterla a un filtro de erosión.

- **Hallar centro de masas:** Después de aplicar los filtros se utiliza una función que ofrece la librería *EmguCV* para encontrar el centro de masas de las partes blancas de la imagen. Esta función reconoce la posición de todos los píxeles de color blanco y halla la media de todas las posiciones, obteniéndose así el centro de masas de la imagen, punto en el que se considera que está la pupila. En la figura 5.19 se puede observar el centro de masas calculado marcado con una cruz.

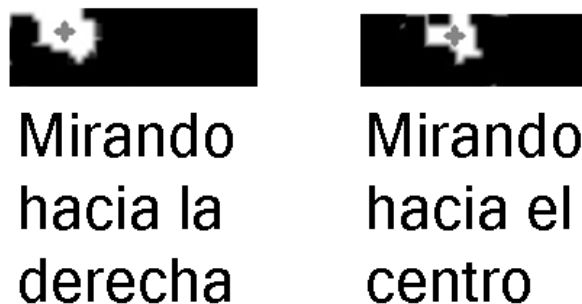


Figura 5.19: Región de interés con centro de masas detectado.

- **Devolver posición del ojo:** Para finalizar el proceso se debe interpretar la posición de la pupila para devolver un resultado que se pueda mostrar por pantalla. Se ha decidido devolver la posición de la pupila como un *String* que indique si el usuario está mirando hacia la derecha, izquierda o centro. Para distinguir entre las posiciones se detecta en qué posición de la imagen se encuentra el centro de masas, si está en la parte derecha de la imagen se considera que el usuario está mirando hacia la derecha.

En la Figura 5.20 se representa el diagrama del proceso completo al que es sometido cada *frame*.

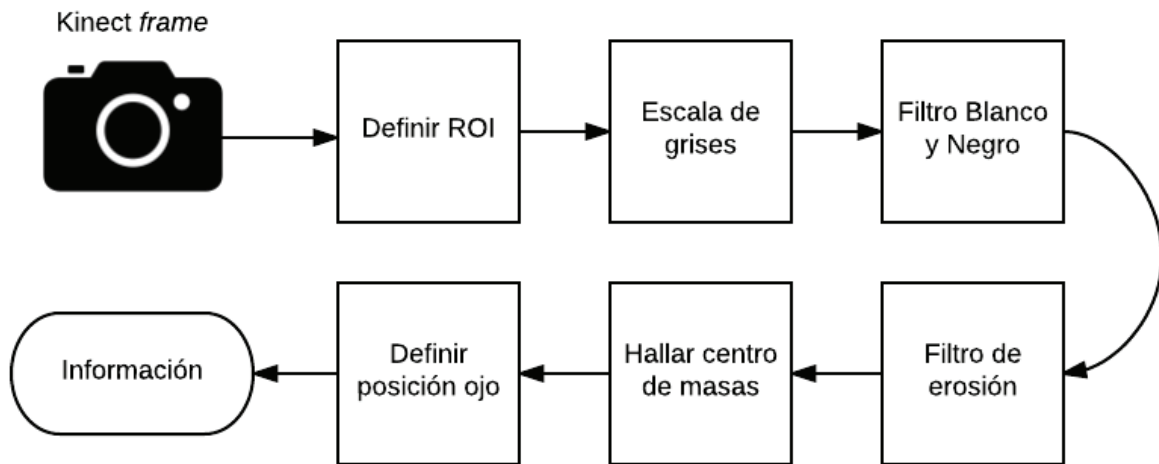


Figura 5.20: Proceso al que se someten los *frames* para detectar la posición de los ojos.

El método que realiza la detección actúa en paralelo al resto de las funciones de la aplicación, ya que hay que realizar varias operaciones sobre cada fotograma y esto ralentiza el funcionamiento de la aplicación. Además, para optimizar el proceso y conseguir mayor fluidez en la ejecución se ha decidido que el método de detección ocular se ejecuta cada 50 milisegundos. El dispositivo Kinect captura 30 imágenes por cada segundo, por lo que en muchas de las capturas consecutivas la posición de los ojos va a ser exactamente la misma. De esta manera se obtendrá la posición de los ojos solamente 15 veces por cada segundo.

## Sistema de alarmas

Para terminar la monitorización del usuario se propone utilizar la información recogida hasta ahora para crear nueva información, es decir, interpretar los datos recogidos de forma que pueda sacarse información nueva de ellos que puede ser de ayuda. Para ello se ha utilizado un sistema de alarmas. El sistema que se ha utilizado se ha desarrollado en el trabajo fin de máster *Detección del estado emocional de un conductor mediante el uso de imágenes infrarrojas* [33] y se le ha añadido un caso más teniendo en cuenta la posición de los ojos.

Las alarmas se activan dependiendo no sólo de la información del último fotograma, si no también de los fotogramas anteriores. Sin embargo la información recogida en cada fotograma hasta ahora no es fiable al 100 % por lo que pueden existir *frames* en los que la información detectada puede ser errónea y no puede ser tomada en cuenta para la activación de una alarma. Por lo tanto, la información detectada sólo se tiene en cuenta si se encuentra una secuencia grande de imágenes con características similares. Por ejemplo, si se tienen 20 fotogramas de los cuales 14 indican que los ojos están cerrados, entonces se considera esto como cierto sólo si estos se encuentran en orden consecutivo, de lo contrario se consideran falsos positivos. En las figuras 5.21 y 5.22 se muestra lo que se considera como una secuencia de falsos positivos y una secuencia válida respectivamente.

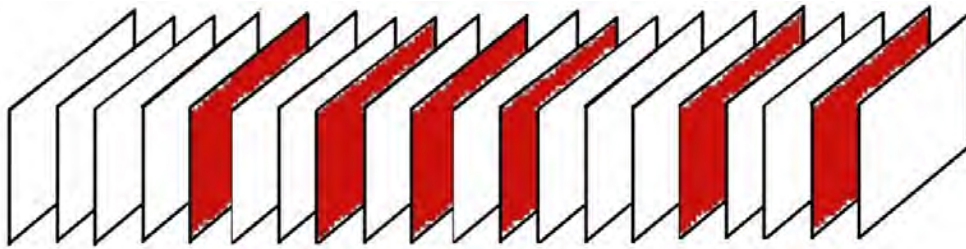


Figura 5.21: Eventos registrados de forma intercalada [33].

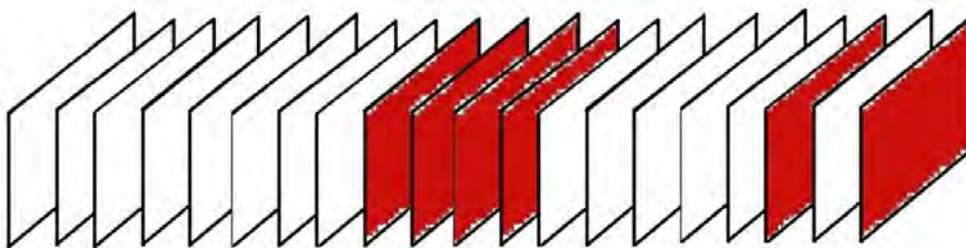


Figura 5.22: Eventos registrados consecutivamente [33].



Para este sistema de alarmas se consideran los siguientes casos que pueden generar situaciones peligrosas:

- ***Pestaños largos***: Si el usuario mantiene los ojos cerrados durante periodos de tiempo demasiado largos.
- ***Boca abierta***: Si el usuario mantiene la boca abierta demasiado tiempo se puede interpretar como un bostezo.
- ***Giro de cabeza***: Si el usuario gira la cabeza y permanece en esta posición durante demasiado tiempo.
- ***Habla***: Si se encuentra una secuencia de *frames* en los que se alterna entre boca abierta y cerrada se puede considerar que el usuario está hablando.
- ***Ingesta de bebidas***: Si el usuario inclina la cabeza hacia arriba y tapa su boca se deduce que puede estar bebiendo, aunque no es demasiado preciso.
- ***Posición de los ojos***: Esta alarma se ha añadido complementando a la alarma de giro de cabeza, mediante la cual sólo se considera que el usuario se ha despistado si su cabeza se mantiene girada, es decir, que un usuario que mantiene su cabeza mirando hacia el frente pero está mirando con los ojos en otra dirección es considerado como una situación segura. Por lo tanto, se utiliza la nueva funcionalidad implementada que detecta la posición de la pupila y se hace saltar la alarma si el usuario está mirando hacia la izquierda o hacia la derecha durante demasiado tiempo.

### 5.1.5. Comunicación mediante sockets

Como ya se ha explicado anteriormente, el sistema se conecta con el simulador de conducción mediante la implementación de un socket síncrono, utilizando una arquitectura cliente-servidor. El equipo en el que se encuentra esta aplicación actúa como servidor y activa un socket mientras espera por la conexión. Una vez que el socket está abierto y el servidor espera para la conexión se puede ejecutar el simulador en el equipo que actúa como cliente. En este momento el cliente envía un mensaje al servidor solicitando que éste envíe información sobre la monitorización del usuario. En el momento en que este emnsaje es recibido, el servidor envía al cliente una línea de texto que contiene la información recopilada por el programa que se ha explicado en los puntos anteriores. La estructura que sigue la línea de texto se expone a continuación:

- ***Smile***: Indica si el usuario sonríe. Puede tomar los valores “Yes”, “No”, “Maybe” o “Unknown”.
- ***Lookingaway***: Indica si el usuario no tiene la cabeza apuntando al frente. Puede tomar los valores “Yes”, “No”, “Maybe” o “Unknown”.
- ***Wearingglasses***: Indica si el usuario está usando gafas. Puede tomar los valores “Yes”, “No”, “Maybe” o “Unknown”.
- ***Lefteyeclosed***: Indica si el usuario tiene el ojo izquierdo cerrado. Puede tomar los valores “Yes”, “No”, “Maybe” o “Unknown”.
- ***Righteyeclosed***: Indica si el usuario tiene el ojo derecho cerrado. Puede tomar los valores “Yes”, “No”, “Maybe” o “Unknown”.



- ***Mouthclosed***: Indica si el usuario tiene la boca cerrada. Puede tomar los valores “Yes”, “No”, “Maybe” o “Unknown”.
- ***Yaw***: Indica el ángulo de una de las tres dimensiones en las que se puede girar la cabeza. Se representa mediante un entero.
- ***Pitch***: Indica el ángulo de una de las tres dimensiones en las que se puede girar la cabeza. Se representa mediante un entero.
- ***Roll***: Indica el ángulo de una de las tres dimensiones en las que se puede girar la cabeza. Se representa mediante un entero.
- ***Gaze***: Indica la posición de las pupilas del usuario. Puede tomar los valores “Centro”, “Izquierda” o “Derecha”.

El simulador solicita información constantemente, de modo que se envía la información recopilada en tiempo real. En la figura 5.23 se puede observar cómo está interconectado el sistema de forma global junto con el simulador.

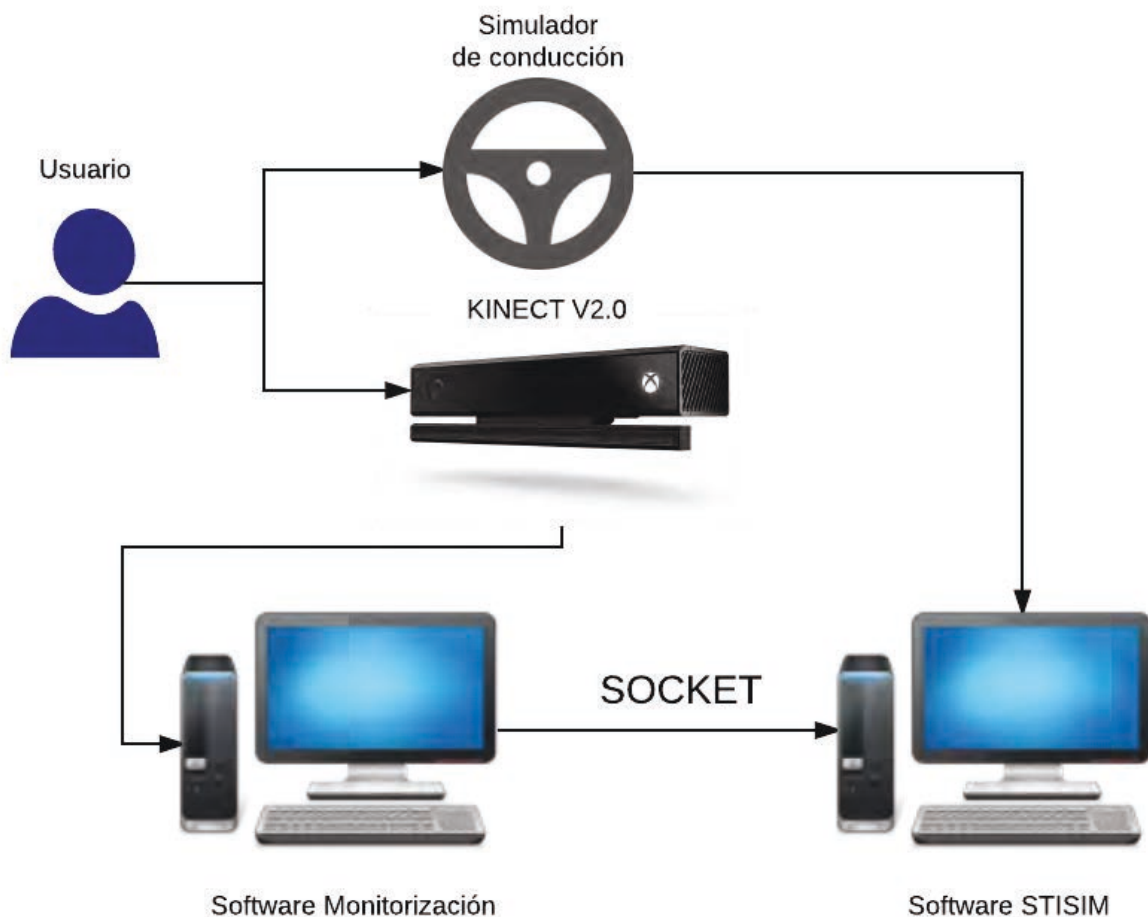


Figura 5.23: Estructura global del sistema.

# Capítulo 6

## Resultados y evaluación

En este capítulo se exponen los resultados obtenidos tras la implementación del sistema tras probar su funcionamiento de diversas formas. Se somete al sistema a una serie de pruebas con distintos sujetos para comprobar su efectividad con cada uno de ellos y poder detectar los escenarios en los que el sistema funciona mejor para tenerlo en cuenta en el futuro. En cada una de las pruebas que se realizan a continuación se comprueba el funcionamiento de cada una de las funcionalidades del sistema y se evalúa su rendimiento.

Para evaluar el funcionamiento de la aplicación se ha escogido a diez sujetos de pruebas de distinto género y distinta apariencia física, de esta forma se observa el comportamiento del sistema al monitorizar distintos sujetos con distintas características. En la figura 6.1 se pueden observar imágenes de los sujetos con los que se ha probado la efectividad del sistema.

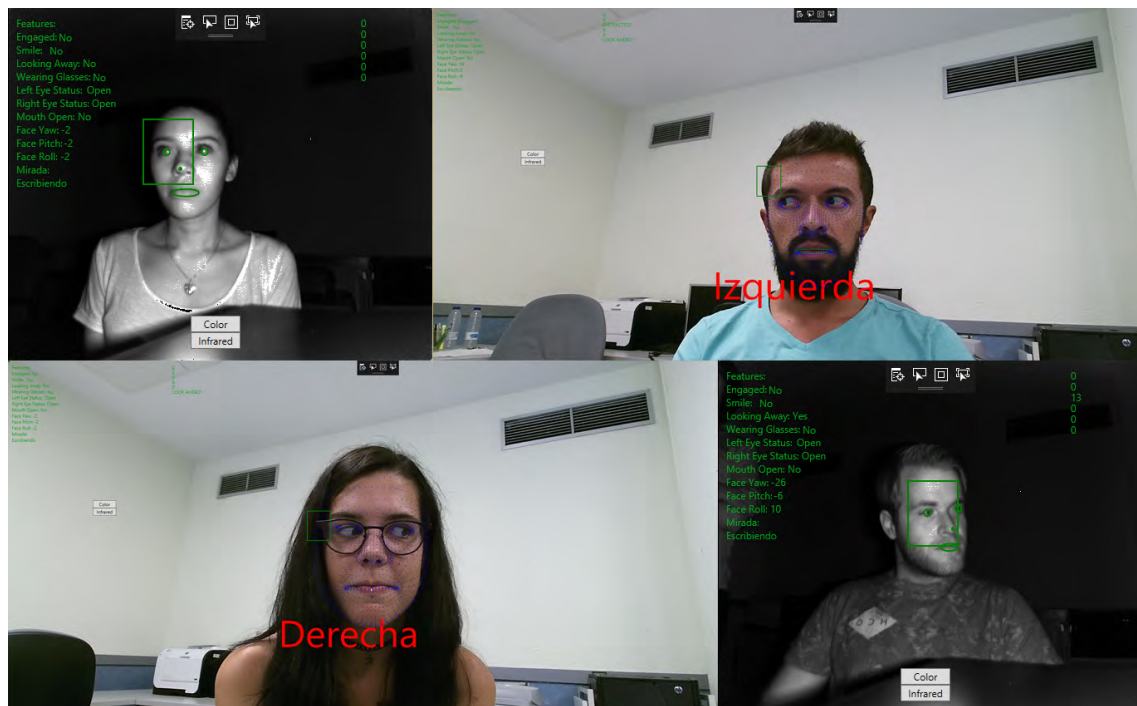


Figura 6.1: Imágenes de los sujetos de pruebas.

A continuación, se exponen los aspectos que se van a tener en cuenta como variables en la realización de las pruebas:

- **Distancia a la cámara:** Se ejecutarán los movimientos que el sistema debe controlar a diferentes distancias de la cámara, para comprobar su eficacia. Se realizarán las pruebas desde cerca, lejos y a una distancia media de la cámara.
- **Luminosidad:** Se probará el funcionamiento del sistema en distintas condiciones de luminosidad, con luz artificial (condición lumínica óptima) y sin ella y con luz natural escasa (peor condición lumínica).
- **Posición de la cabeza:** La posición de la cabeza del usuario influye en la detección de la posición de las pupilas.
- **Género y apariencia:** Se realizan las pruebas con sujetos masculinos y femeninos con cualidades físicas diferentes para comprobar cómo pueden influir en el funcionamiento del sistema.

A continuación se exponen los resultados observados tras probar todas las funcionalidades sometiendo al sistema a las variables anteriormente mencionadas.

## 6.1. Reconocimiento facial

Para empezar se prueba la cámara infrarroja realizando movimientos de cabeza en todas las direcciones, abriendo y cerrando los ojos y la boca. Todas las actividades son detectadas por la cámara sin problemas, la posición de la cabeza es muy precisa dentro del rango de movimientos que permite. Sólo se pierde la posición cuando el usuario gira la cabeza completamente hacia arriba ocultando su cara, pero una vez que se vuelve a bajar la cabeza el dispositivo vuelve a detectarla al instante. El resto de características de la cara también se detectan fácilmente siempre que la posición de la cabeza no impida verlas. Si este es el caso, la situación de ojos y boca aparece como desconocida.

Se exponen a continuación los resultados obtenidos al variar los parámetros anteriores.

### Distancia

Una vez que el cuerpo y la cara del usuario han sido localizados la distancia a la cámara no influye para nada en el resultado final, siempre que el usuario siga apareciendo en pantalla. Se ha observado que para conseguir localizar el cuerpo del usuario a veces es necesario alejarse del dispositivo, pero una vez que se realiza la localización la distancia no es un problema. También hay que tener en cuenta que este sistema está pensado para utilizar dentro de un vehículo, por lo que no sería necesario considerar las distancias largas para un buen funcionamiento.

## Luminosidad

Como se está utilizando la cámara de infrarrojos, la luminosidad no influye para nada en el rendimiento del programa, no importa si se utiliza luz artificial, natural o se está completamente en la oscuridad, ya que la cámara proyecta su propio haz de luz infrarroja.

## Género y apariencia

De entre todas las personas con las que se han realizado las pruebas, sólo se ha observado un caso en el que el sistema puede fallar, si el pelo del usuario tapa parte de su cara, aunque los ojos estén visibles, puede verse perjudicado la detección de la apertura de los ojos. Evidentemente si se obstruye la visión de la cara del usuario, el sistema no va a funcionar correctamente, pero aunque la obstrucción sea en zonas no vitales para la detección facial (como la frente, cejas, etc...) es posible que el dispositivo malinterprete ciertas posiciones y devuelva resultados erróneos.

También se ha probado el sistema con usuarios con gafas y estas son detectadas sin problemas, aunque estas pueden crear un reflejo que entorpezcan la detección de los ojos en algún momento.

A continuación, en la tabla 6.1 se muestran los resultados generales de las pruebas de reconocimiento facial utilizando la cámara infrarroja:

Tabla 6.1: Resultados del reconocimiento facial con la cámara Infrarroja

Parte a detectar	Distancia			Luminosidad			Género	
	Cerca	Media	Lejos	Alta	Media	Baja	Masculino	Femenino
Ojos	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado
Boca	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado
Posición Cabeza	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado	Detectado

## 6.2. Reconocimiento Ocular

Esta funcionalidad presenta más problemas que la anterior, debido a que está sujeta a las condiciones lumínicas y es muy dependiente de éstas. Aun así, el comportamiento del sistema es similar en todos los usuarios con los que se ha probado. El mayor problema observado era el posicionamiento erróneo de los puntos que rodean los ojos dependiendo de las características de la cabeza del usuario. No obstante, este problema se subsanó utilizando una combinación entre los puntos detectados con la cámara de infrarrojos y la cámara de alta definición, de manera que el sistema localiza la posición del ojo sin problemas.

A continuación se muestra el comportamiento del sistema ante las distintas variables.

## **Distancia**

En este caso la distancia a la cámara es un factor muy importante a la hora de obtener buenos resultados. Para conocer la posición de la pupila se recorta” la porción de la imagen perteneciente al ojo y esta porción de imagen es con la que se trabaja. El problema ocurre al variar la distancia. Si el usuario está cerca de la cámara, la porción de la imagen ocupada por el ojo es mucho mayor y por lo tanto se tiene una imagen con muchos más píxeles y por lo tanto mucho más nítida, lo que ayuda a identificar la posición de la pupila. Por el contrario, si el usuario está alejado de la cámara, la zona perteneciente al ojo está compuesta por pocos píxeles, por lo que es una imagen mucho más pequeña y menos nítida, lo que ocasiona que, a más distancia, se obtengan peores resultados.

Sin embargo, algunos de los resultados obtenidos a mayores distancias seguían siendo precisos, y sabemos que la distancia a la que se va a utilizar la cámara dentro del coche nunca va a ser demasiado grande.

## **Luminosidad**

Este es el factor que más afecta al funcionamiento del reconocimiento ocular, dado que la posición de la pupila se deduce basándose en las diferencias de oscuridad entre la parte blanca del ojo y la pupila y el iris. Por eso mismo, cuando la luz ambiente es escasa, las zonas oscuras no se diferencian demasiado de las zonas claras, lo que ocasiona un mal funcionamiento. A partir de cierto grado de oscuridad el sistema es inservible.

## **Posición de la cabeza**

Esta funcionalidad se hizo pensando en posiciones de la cabeza más bien centradas, para controlar las situaciones en las que el usuario, pese a estar posicionado de frente a la cámara, está mirando hacia otro lado, lo que puede causar situaciones peligrosas. Por lo tanto, para giros de cabeza muy acentuados, los resultados obtenidos no son exactos y pueden variar bastante.

Como se ha mencionado antes, esta función funciona mejor cuanto más cerca esté la cámara del usuario, pero esto sólo se cumple si la cámara está situada a una altura similar a la cabeza de este. Si el usuario se acerca a la cámara pero esta se encuentra a, por ejemplo, la altura de su pecho, el ángulo formado entre los ojos del usuario y la cámara puede deformar la imagen de modo que los resultados obtenidos sean erróneos.

## **Género y apariencia**

En este caso la funcionalidad no se ve afectada por la apariencia del sujeto siempre y cuando los ojos estén completamente a la vista y puedan ser detectados por la cámara. Como caso excepcional, si se utilizan gafas pueden crearse reflejos que distorsionen la imagen del ojo y, por tanto, alteren los resultados obtenidos.

A continuación, en la tabla 6.2 se muestran los resultados generales de las pruebas de reconocimiento ocular utilizando la cámara de alta definición.

Tabla 6.2: Resultados del reconocimiento ocular con cámara de alta definición

Posición de la cabeza	Distancia			Luminosidad			Género	
	Cerca	Media	Lejos	Alta	Media	Baja	Masculino	Femenino
Centrada	Detectado	Detectado	Con Problemas	Detectado	Con Problemas	No Detectado	Detectado	Detectado
Mirando hacia los lados	Detectado	Con Problemas	No detectado	Detectado	Con Problemas	No Detectado	Detectado	Detectado
Mirando hacia arriba o abajo	No Detectado	No Detectado	No Detectado	No Detectado	No Detectado	No Detectado	No Detectado	No Detectado

### 6.3. Alarmas

Este último caso está sujeto a las mismas condiciones que los dos anteriores, dado que se basa en la información recogida por ambas cámaras, de modo que no es necesario señalar cómo afectan las variables a las alarmas. Simplemente es necesario mencionar que las todas las alarmas utilizan la información obtenida por la cámara de infrarrojos excepto la última, que se basa en los datos obtenidos por la cámara de alta definición.

A continuación se exponen las observaciones realizadas al uso de cada una de las alarmas.

#### Pestañeos largos

Esta alarma funciona perfectamente, ya que se basa en la información sobre los ojos captada por la cámara infrarroja y ésta es muy fiable, de modo que la alarma se activa inmediatamente tras detectar demasiados *frames* con los ojos cerrados.

#### Boca abierta

Esta alarma detecta si el usuario está bostezando y funciona del mismo modo que la anterior. Por lo tanto su funcionamiento no presenta ningún problema, ya que trabaja con información precisa.

#### Giro de cabeza

Esta alarma controla la posición de la cabeza para avisar al usuario en caso de que se despiste. La posición de la cabeza es detectada por la cámara infrarroja con una efectividad similar a los casos anteriores, de modo que esta alarma también funciona correctamente en todos los casos.

## Habla

Esta es una de las alarmas que presentan algunos problemas, ya que tiene en cuenta *frames* consecutivos en los que la boca esté abierta y cerrada. La mayoría de las veces que el usuario está hablando esta alarma comienza a contar el número de *frames*, pero no siempre llegan a acumularse suficientes como para que salte la alarma, ya que el habla es algo irregular.

## Ingesta de bebidas

La efectividad de esta alarma ha resultado ser casi nula, ya que simplemente con levantar la cabeza es probable que existan problemas al detectar la boca y la alarma salta automáticamente. Por lo tanto, no sirve para reconocer adecuadamente si un usuario está bebiendo.

## Posición de los ojos

Para finalizar, esta alarma utiliza la posición de las pupilas captada por la cámara de alta resolución. Como ya se mencionó, esta funcionalidad se ve afectada por la luminosidad, la distancia a la cámara y la posición de la cabeza. Aun así, teniendo en cuenta los casos en los que funciona correctamente, la alarma cumple siempre su cometido, saltando cuando se detecta una cantidad elevada de *frames* en los que los ojos están mirando a derecha o izquierda. Por lo tanto, esta alarma también funciona correctamente, pero está condicionada al funcionamiento de la detección de la pupila, que, se ha visto, no siempre funciona correctamente.

A continuación se muestra la matriz de trazabilidad entre las pruebas descritas anteriormente y los requisitos del sistema en la tabla 6.3.

Tabla 6.3: Matriz de trazabilidad entre pruebas y requisitos

Requisitos del sistema	Pruebas con cámara infrarroja	Pruebas con cámara de alta definición	Pruebas con alarmas
RF-01	X		
RF-02		X	
RF-03	X		
RF-04	X		
RF-05	X		
RF-06	X		
RF-07			X
RF-08			X
RF-09			X
RF-10			X
RF-11			X
RF-12			X
RF-13	X	X	X
RF-14	X	X	X
RF-15	X	X	X
RF-16	X	X	X
RNF-01	X		
RNF-02		X	
RNF-03	X	X	X
RNF-04	X	X	X
RNF-05	X	X	X
RNF-06	X	X	X

## 6.4. Pruebas en el simulador

El objetivo final de este proyecto es utilizar el sistema de monitorización junto con el simulador de conducción, de este modo, el comportamiento del usuario es detectado mientras conduce. Por eso mismo es necesario situar la cámara en el simulador en una posición que no interrumpa la visión de la pantalla y desde la que se pueda observar claramente al usuario. El sitio escogido para la cámara dentro del simulador se muestra en la Figura 6.2. En esta posición se han probado diferentes alturas de la cámara hasta encontrar la óptima.



Figura 6.2: Cámara *Kinect* colocada en el simulador (señalada con círculo rojo).

Sin embargo, se han encontrado ciertos problemas a la hora de situar la cámara en el simulador. Se han observado dificultades para detectar la posición del usuario. Tras cambiar varias veces la posición de la cámara se ha comprobado como el problema no radica en la cercanía del usuario a ella o en la posición desde la que se graba, ya que posiciones similares han funcionado en un entorno ajeno al simulador. Por lo tanto, existe un problema en el entorno del simulador que no permite la detección correcta del cuerpo para comenzar la monitorización.

Se ha podido visualizar el entorno del simulador tal como lo observa la cámara al ser iluminado por el haz de luz infrarroja. En la Figura 6.3 se puede visualizar la captura de la imagen mediante rayos infrarrojos, en la que los objetos del entorno están representados en distintos colores dependiendo de la profundidad a la que se encuentren, es decir, la distancia a la cámara. Los objetos situados a menor distancia de la cámara aparecen en la imagen en tonos más cálidos, mientras que los objetos situados más lejos están representados con tonalidades más frías.



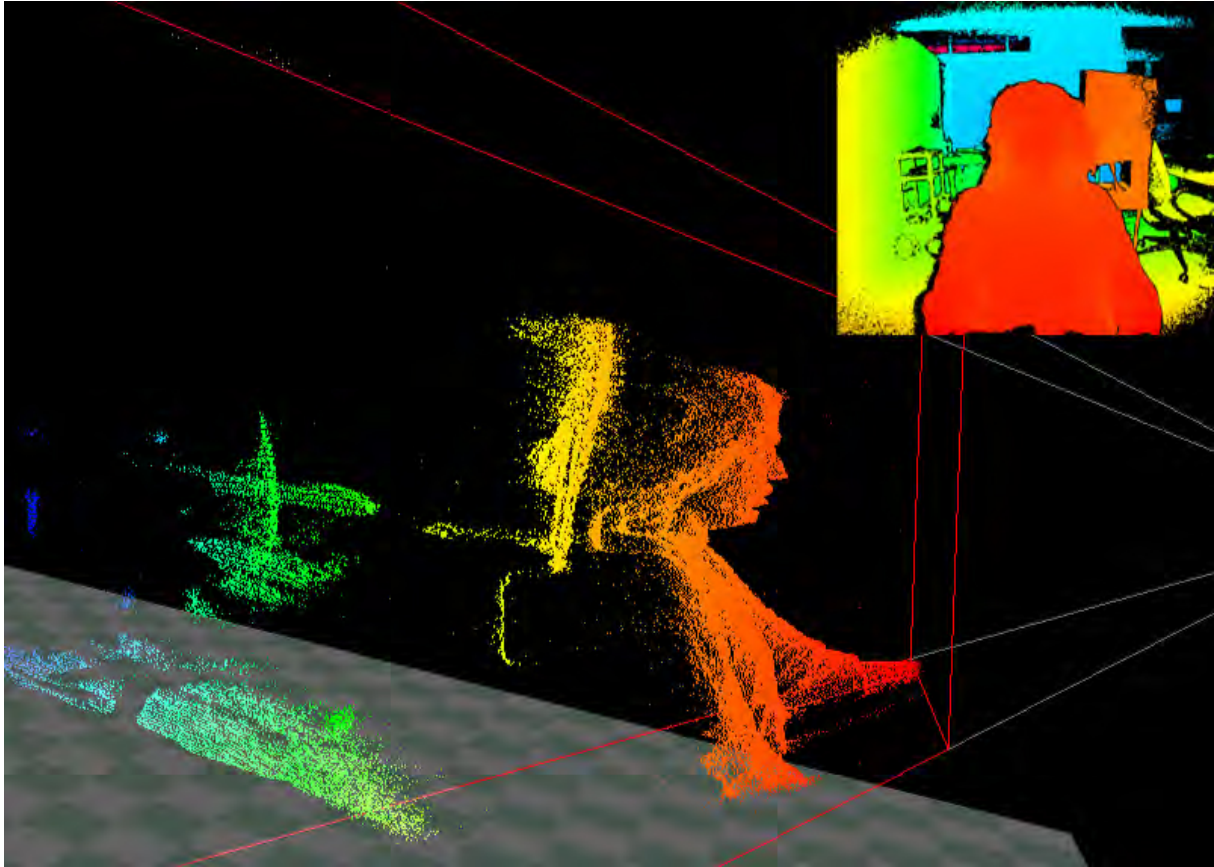


Figura 6.3: Mapa de profundidad del entorno del simulador.

Comprobando esta captura, se ha observado que, tanto el asiento como el monitor del equipo pertenecientes al simulador de conducción, aparecen en una zona de color similar al cuerpo de la persona que está utilizándolo. En la figura 6.3 se puede ver, además, cómo el cuerpo del usuario aparece unido a estos objetos cercanos. Todo esto lleva a pensar que el motivo para que no se reconozca al usuario de forma correcta puede ser que los objetos por los que está formado el simulador están interfiriendo en la detección del cuerpo del usuario. El dispositivo Kinect detecta estas partes del simulador junto con la persona como si fueran parte del mismo objeto, lo que lleva a no distinguir bien la figura de los hombros y la cabeza del usuario y no poder detectarlo correctamente.

Por lo tanto, se ha llegado a la siguiente solución. Para poder utilizar el sistema de monitorización junto con el simulador de conducción, es necesario ser detectado por la cámara antes de sentarse en el simulador, de manera que el entorno de éste no distorsione la percepción del cuerpo por parte de la cámara. Una vez que el cuerpo es detectado, el usuario puede sentarse en el simulador y el sistema seguirá detectándole siempre que no desaparezca del campo de visión de la cámara o la visión del cuerpo del usuario sea obstruida mediante ningún objeto. De este modo, se puede utilizar la cámara dentro del entorno del simulador, el cual es el objetivo principal de este proyecto.

# Capítulo 7

## Planificación del trabajo y entorno socioeconómico

En este capítulo se explicarán varios puntos sobre el desarrollo del proyecto. Primero se describirá la metodología software que se ha seguido durante el proceso de implementación. A continuación se expone la planificación de trabajo, seguida de la estimación de costes tanto materiales como de personal que forman parte del presupuesto necesario para el desarrollo del proyecto en su totalidad. Para finalizar se analiza el impacto que puede generar el sistema en diferentes ámbitos de la sociedad

### 7.1. Metodología de desarrollo Software

A la hora de realizar un proyecto, es necesario fijar una metodología a seguir durante todo el desarrollo, de esta manera se fijan objetivos que pueden ayudar a la realización del trabajo desde distintos enfoques. La metodología *representa el camino para desarrollar software de una manera sistemática*” [35]. Para este proyecto, en un primer lugar, se pensó en seguir un modelo iterativo y creciente, también conocido como modelo incremental, el cual consiste en fijar una sucesión de etapas a seguir durante el desarrollo (conocidas como modelo en cascada) y realizar iteraciones sobre el mismo proceso para añadir nuevas funcionalidades al sistema. Sin embargo, este proyecto cuenta con unos objetivos claros desde el inicio, por lo que no será necesaria la repetición del ciclo de desarrollo, además de que esto añadiría a la planificación más complejidad de la necesaria. En su lugar, se seguirá el modelo en cascada correspondiente a una sola iteración, teniendo la posibilidad de volver atrás en cada una de las etapas que componen la secuencia en caso de encontrar algún error que necesite revisarse.

El modelo en cascada fue presentado por primera vez por Winston W. Royce en 1970, aunque sufrió varias revisiones en los años posteriores por las que es más conocido [36]. Como se ha expuesto anteriormente, en este modelo el producto evoluciona durante etapas que se suceden linealmente, permitiendo volver a la etapa anterior si se considera necesario. El número de etapas puede variar, pero suele seguir el mismo patrón. Más adelante, en la figura 7.1, se muestran las fases que componen el modelo utilizado para este proyecto:

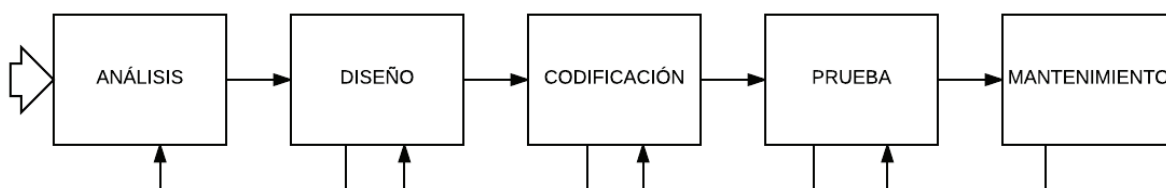


Figura 7.1: Diagrama de la metodología en cascada.

A continuación se pasa a explicar individualmente cada una de las fases vistas en la figura :

- **Análisis:** Esta la fase inicial en la que se fijan los objetivos finales a partir de las necesidades del usuario. Se realiza una lista de requisitos que especifican estos objetivos durante reuniones entre el usuario y las personas a cargo del proyecto.
- **Diseño:** Una vez se ha realizado la fase de análisis y se conocen los objetivos del proyecto comienza la fase de diseño, en la que se desarrolla una arquitectura compuesta por módulos que representan distintas funcionalidades del sistema.
- **Codificación:** Tras diseñar el sistema, éste debe ser implementado en código.
- **Prueba:** Después de implementar el sistema hay que realizar pruebas sobre el código para corroborar que se cumplen los requisitos fijados durante el análisis.
- **Mantenimiento:** Se debe verificar el correcto funcionamiento del sistema por parte del usuario final, pudiendo solucionar problemas en caso de que éste encontrara alguno.

Además, como añadido a las fases anteriores se añade una constante investigación que puede realizarse durante cualquier fase. Además, una vez terminadas todas las etapas, se debe documentar todo el proceso.

## 7.2. Planificación

A continuación se expone y detalla la planificación del trabajo fin de grado y cómo se ha gestionado el tiempo para su realización. El periodo de trabajo comenzó el 01 de Febrero de 2016 y terminó el 21 de septiembre de 2016 tal como se muestra en la tabla 7.1.

Tabla 7.1: Tabla general de planificación del proyecto.

Trabajo	Duración (horas)	Fecha de comienzo	Fecha de finalización
Desarrollo del proyecto	270	20/01/2015	05/08/2016
Tarea de documentación	61	06/08/2016	21/09/2016

De este modo, el proyecto se ha completado en 331 horas. Hay que destacar diferentes intervalos en los que se detuvo el trabajo por asuntos académicos, como el periodo del 20 de mayo al 1 de junio, o por asuntos personales desde el 10 hasta el 17 de agosto. El resto de semanas se ha trabajado cada día un número variable de horas de lunes a viernes, dejando los fines de semana como descanso.

Ésta es la planificación global, a continuación se detalla el tiempo utilizado para cada una de las fases y se aportan datos más específicos sobre cada una de ellas. En la tabla 7.2 se puede ver la etapa de desarrollo, la duración en horas que ha supuesto y la fecha de inicio y de final de cada proceso.

Tabla 7.2: Planificación.

<b>Etapa</b>	<b>Duración (horas)</b>	<b>Fecha de comienzo</b>	<b>Fecha de finalización</b>
<b>ANÁLISIS DEL SISTEMA</b>	<b>25</b>	<b>20/01/2016</b>	<b>11/02/2016</b>
Reuniones iniciales con tutor	7	20/01/2016	01/02/2016
Especificación de requisitos	7	25/01/2016	03/02/2016
Casos de uso	6	03/02/2016	11/02/2016
<b>INVESTIGACIÓN</b>	<b>60</b>	<b>11/02/2016</b>	<b>04/03/2016</b>
<b>DISEÑO DEL SISTEMA</b>	<b>35</b>	<b>07/03/2016</b>	<b>05/04/2016</b>
Definición de módulos	12	07/03/2016	21/03/2016
Interacción entre módulos	18	21/03/2016	05/04/2016
<b>IMPLEMENTACIÓN DEL SISTEMA</b>	<b>130</b>	<b>06/04/2016</b>	<b>29/07/2016</b>
Gestión de <i>frames</i>	27	06/04/2016	18/04/2016
Interfaz Gráfica	7	19/04/2015	22/04/2015
Socket	13	25/04/2016	02/05/2016
Alarmas	5	02/05/2016	06/05/2016
Seguimiento ocular	78	09/05/2016	29/07/2016
<b>PRUEBAS DEL SISTEMA</b>	<b>20</b>	<b>29/07/2016</b>	<b>05/08/2016</b>
Especificación de las pruebas	9	29/07/2016	31/07/2016
Ejecución de las pruebas	11	01/08/2016	05/08/2016

Para finalizar se vuelve a mostrar la planificación de trabajo expuesta en las tablas anteriores pero esta vez mediante un diagrama de Gantt (figura 7.2), para presentar los datos de manera más visual y facilitar su entendimiento.

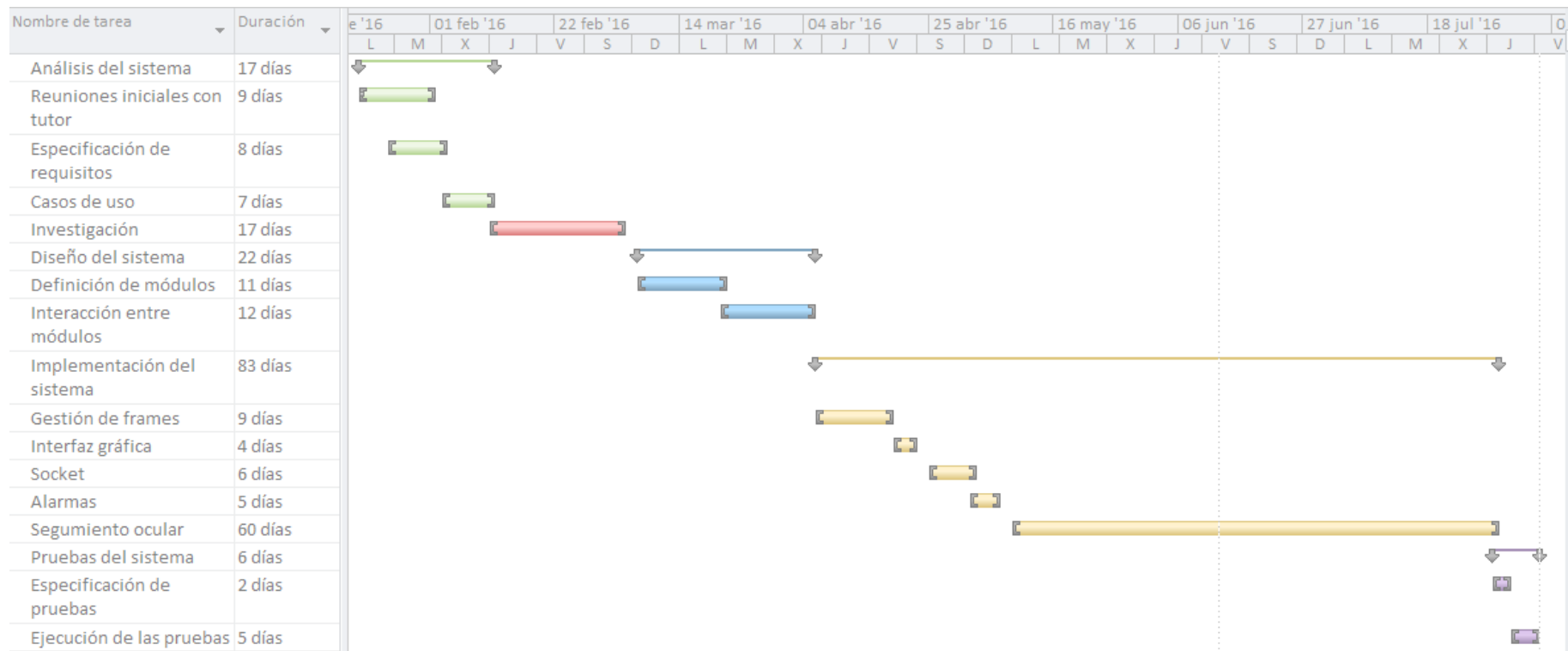


Figura 7.2: Planificación Gantt del proyecto.

## 7.3. Presupuesto

En esta sección se estiman los costes que se producirían en caso de llevar a cabo este proyecto en el contexto de una empresa. Primero se analizarán los costes materiales y de personal que se han considerado y finalmente los costes globales del proyecto.

### 7.3.1. Costes Parciales

#### Costes materiales

Los costes materiales hacen referencia al dinero dedicado a los productos utilizados durante el desarrollo del proyecto. Para realizar la estimación del coste es necesario conocer el precio de estos productos, el tiempo durante el que se han utilizado y su periodo de amortización. Toda esta información se muestra en la tabla 7.3.

Tabla 7.3: Tabla de costes de material.

Producto	Precio (€)	Periodo amortización (meses)	Tiempo uso (meses)	Coste en proyecto (€)
PC simulador	900,08	48	2	37,5
PC desarrollo	1220,00	48	7	177,91
STI SIM M100	7.500,00	48	2	312
KINECT 2.0	149,99	48	7	21,87
<b>Total</b>	-	-	-	<b>549,28</b>

El coste de cada producto se ha calculado dividiendo su precio entre su periodo de amortización y multiplicando el resultado por el tiempo de uso.

#### Costes de personal

Además de los costes materiales, también hay que tener en cuenta los costes de personal, formados por los sueldos de cada una de las personas necesarias para llevar a cabo el proyecto. Como mínimo es necesario considerar un puesto de trabajo para cada una de las etapas estipuladas en la planificación según el modelo en cascada. Por lo tanto, los roles definidos para este proyecto son:

- Analista: Persona encargada de obtener los requisitos del proyecto.
- Diseñador: Persona encargada de diseñar el sistema y su arquitectura
- Programador: Persona encargada de implementar el código del sistema.
- Técnico de pruebas y mantenimiento: Persona encargada de comprobar el funcionamiento del sistema mediante pruebas.
- Jefe de proyecto: Encargado de dirigir el desarrollo del proyecto y coordinar al resto del personal.
- Investigador: Persona encargada de la documentación de todo el proyecto.

Cada uno de estos roles tiene asociado un sueldo por hora, los cuales se han obtenido de la tabla salarial publicada el 30 de enero de 2015 por la Dirección General de Empleo y se exponen junto con el coste total en la tabla 7.4.

Tabla 7.4: Tabla de costes de personal.

Rol	Coste por hora (€)	Horas de trabajo	Total (€)
Jefe de proyecto	25,47	12	305,64
Analista	24,39	25	609,75
Diseñador	24,39	35	853,65
Programador	18,94	130	2.462,2
Técnico de pruebas	18,94	20	378,8
Investigador	16,04	121	1.940,84
Total	-	343	6.550,88

### 7.3.2. Coste Total

El coste total no es simplemente la suma de los costes parciales anteriores, también hay que tener en cuenta costes indirectos, impuestos, margen de riesgo y margen de beneficio. Los porcentajes que se han considerado son los siguientes:

- Coste indirecto: 10 %. Costes de internet, teléfono, electricidad, etc.
- Margen de riesgo: 15 %. Margen para cubrir perdidas imprevistas.
- Beneficios: 15 %. Beneficios generados por el proyecto.
- IVA: 21 % Impuesto sobre el Valor Añadido.

Tabla 7.5: Tabla de costes totales.

Concepto	Cantidad (€)
Total costes directos	7.100,16
Costes indirectos (10 %)	710,02
Margen de riesgo (15 %)	1.065,02
Beneficios (15 %)	1.065,02
Base imponible	9.940,22
IVA (21 %)	2.087,45
Total coste del proyecto	12.027,67

## 7.4. Impacto del proyecto

En esta sección se trata el posible impacto del proyecto en distintos ámbitos de la sociedad. Para empezar, en el ámbito socioeconómico, este proyecto se encuentra dentro del sector del automovilismo, ya que su objetivo es formar parte de un Sistema Avanzado de Asistencia a la Conducción. Los sistemas de asistencia actuales son de distintos tipos, algunos son más básicos y podemos encontrarlos en la mayoría de los coches del mercado, pero otros son sistemas muy complejos por los que se pide un precio más alto. En este caso, se trata del segundo grupo. El sistema multiagente completo utilizando también el sistema de monitorización subiría bastante el precio final del vehículo, aunque con el tiempo podría acabar amortizándose, al evitar más accidentes se evitan reparaciones innecesarias. También hay que tener en cuenta que en el futuro el coste de los materiales será menor y el diseño del sistema estará optimizado, por lo que podría disminuir el coste del vehículo aún más.

La reducción de accidentes de tráfico no sólo supone un beneficio económico para los usuarios de estos sistemas, también supone un ahorro de dinero para el gobierno, ya que en muchos de estos accidentes se dañan infraestructuras públicas. Aun así, la reducción de accidentes podría causar un efecto negativo en los negocios de reparaciones, los cuales tendrán menos clientes. Lo mismo pasa con los concesionarios, aunque ganen dinero vendiendo esta nueva tecnología, no venderán tantos coches como antes, ya que muchos de los vehículos son comprados a causa de tener que renovar el coche por un accidente de tráfico.

Dentro del ámbito social, esta tecnología supone un gran impacto positivo, dado que puede evitar una gran cantidad de desgracias y salvar vidas humanas. Como ya se ha mencionado, el mayor número de muertes anuales son causadas por accidentes de tráfico.

Para finalizar, el impacto medioambiental también es positivo. Durante los accidentes se liberan a la atmósfera una gran cantidad de residuos y gases perjudiciales. Además, como ya se ha dicho antes, se reduce el número de coches desechados por accidentes, los cuales se seguirán utilizando. Esto a la larga genera un impacto medioambiental muy positivo.



# Capítulo 8

## Conclusiones y trabajos futuros

En este capítulo final se exponen las conclusiones a las que se ha llegado tras la implementación y desarrollo del trabajo fin de grado, así como las líneas de trabajo futuras hacia las que se puede dirigir este proyecto.

### 8.1. Conclusiones técnicas

Una vez que se ha terminado el proyecto, es necesario realizarse la siguiente pregunta: “¿Se han cumplido los objetivos establecidos al comienzo del mismo?”. En este caso se puede concluir que estos objetivos principales sí han sido cumplidos.

Se ha conseguido realizar la monitorización del usuario y controlar en todo momento la posición y situación de diferentes rasgos de la cara a tener en cuenta. Se ha encontrado una forma de recopilar esta información y mostrarla por pantalla. También se ha conseguido un método fiable para llevar a cabo el reconocimiento de estas características independientemente de las condiciones lumínicas mediante el uso de una cámara infrarroja. Además se ha generado información adicional basándose en los datos iniciales mediante el sistema de alarmas.

Como añadido se ha implementado el sistema de reconocimiento de la posición de las pupilas del usuario, característica que el dispositivo Kinect v2.0 no es capaz de detectar. Esta nueva funcionalidad puede ser utilizada en condiciones lumínicas favorables.

Otro de los objetivos de este proyecto era establecer la comunicación entre el sistema que controla la monitorización y el sistema donde se ejecuta la simulación de conducción. Este objetivo también se ha cumplido mediante la implementación de un socket síncrono que conecta los equipos que alojan estos sistemas y permite el intercambio de mensajes entre ellos.

Como conclusión general, se han aplicado de forma satisfactoria los conocimientos adquiridos durante el grado a un problema relativo a la ingeniería informática.

## 8.2. Conclusiones personales

Durante la realización de este trabajo se ha tomado conciencia de la magnitud y dificultad que lleva consigo un proyecto de estas dimensiones, teniendo que fijar los objetivos principales y planificando los tiempos de diseño, implementación, etc. También se ha tomado contacto con el trabajo llevado a cabo por el grupo de investigación CAOS de esta universidad, ayudando a desarrollar una pequeña parte de un proyecto mucho mayor. Durante el proceso se han adquirido ciertas habilidades de trabajo y aprendizaje necesarias para el futuro.

También vale la pena mencionar los conocimientos adquiridos acerca de la monitorización de usuarios y captura de movimiento mediante dispositivos como el que se ha usado durante el proyecto, aprendiendo a desenvolverse con las herramientas necesarias para su manipulación, así como los conocimientos sobre procesamiento de imágenes en un sistema que funciona en tiempo real.

## 8.3. Trabajos futuros

El sistema desarrollado en este trabajo de fin de grado es una primera aproximación a la monitorización de un conductor de un vehículo y únicamente se controlan información bruta sobre el estado físico del usuario. En el futuro se puede profundizar en el ámbito de las emociones, las cuales, como ya se ha mencionado, tienen un impacto muy grande en la manera de comportarse de las personas y pueden influir perjudicialmente en la conducción. Mediante una detección facial más detallada se podrían llegar a tener en cuenta estas emociones en la conducción. Posiblemente se podría utilizar la información captada por la cámara de alta definición del dispositivo Kinect v2.0, aunque es posible que la expresiones faciales no sean lo suficientemente detalladas.

El paso a seguir a continuación de este trabajo sería la integración de la información captada por este sistema dentro del sistema multiagente desarrollado por el departamento CAOS. Esta información del conductor deberá ser utilizada dentro de la ontología del sistema multiagente y así éste podrá considerar situaciones mucho más complejas durante la simulación de conducción.

El otro aspecto a mejorar es el seguimiento de la posición de la pupilas, este se controla de forma independiente de la posición de la cabeza y sólo considera tres valores para la posición del ojo. Como trabajo futuro se propone encontrar otro método que permita conocer la posición de la pupila de manera precisa y la dirección de la mirada teniendo en cuenta la posición de la cabeza. Este método, además, para ser todavía más preciso deberá funcionar con las imágenes infrarrojas, para no depender de la iluminación y poder ser utilizado en cualquier situación mientras se conduce.

# Bibliografía

- [1] OMS, *Informe sobre la situación actual de la seguridad vial*, 2015
- [2] DIRECCIÓN GENERAL DE TRÁFICO, *Otros factores de riesgo: La fatiga*. URL: [http://www.dgt.es/PEVI/documentos/catalogo\\_recursos/didacticos/did\\_adultas/fatiga.pdf](http://www.dgt.es/PEVI/documentos/catalogo_recursos/didacticos/did_adultas/fatiga.pdf) (Último acceso 15/09/2016).
- [3] CENTRO ZARAGOZA. INSTITUTO DE INVESTIGACIÓN SOBRE VEHÍCULOS, S.A., *Los sistemas de detección de la fatiga del conductor*, 2009. URL: [http://www.centro-zaragoza.com:8080/web/sala\\_prensa/revista\\_tecnica/hemeroteca/articulos/R42\\_A7.pdf](http://www.centro-zaragoza.com:8080/web/sala_prensa/revista_tecnica/hemeroteca/articulos/R42_A7.pdf) (Último acceso 15/09/2016).
- [4] GERMÁN GUTIERREZ, JOSÉ ANTONIO IGLESIAS, FRANCISCO JAVIER ORDOÑEZ, AGAPITO LEDEZMA, ARACELI SANCHIS, *Agent-Based Framework for Advanced Driver Assistance Systems in Urban Environments*, 2014.
- [5] AUTOCONNECTEDCAR.COM, *ADAS Definition*, 2012. URL: <http://tinyurl.com/j8qe6ce> (Último acceso: 15/09/2016).
- [6] SYSTEMS TECHNOLOGY, INC., *STISIM Drive: Car Driving Simulator & Simulation Software*. URL: <http://www.stisimdrive.com/products/simulation-systems/m100-series> (Último acceso: 29/03/2016).
- [7] CANALMOTOR, *Control de cruce adaptativo, una ayuda muy útil*, 2015. URL: <http://www.motor.mapfre.es/consejos-practicos/seguridad-vial/4124/control-de-cruce-adaptativo-una-ayuda-muy-relajante> (Último acceso 15/09/2016).
- [8] CAPREOLUS, *¿Qué es el AEB o frenada de emergencia autónoma?*, 2012. URL: <http://www.circulaseguro.com/que-es-el-aeb-o-frenada-de-emergencia-autonoma/> (Último acceso 15/09/2016).
- [9] IBÁÑEZ, *Asistente de mantenimiento en carril: ¿comodidad o seguridad?*, 2013. URL: <http://www.circulaseguro.com/asistente-de-mantenimiento-en-carril-comodidad-o-seguridad/> (Último acceso 15/09/2016).
- [10] IBÁÑEZ, *Así funciona el asistente para atascos de tráfico del nuevo Volkswagen Passat*, 2015. URL: <http://www.motorpasion.com/tecnologia/asi-funciona-el-asistente-para-atascos-de-trafico-del-nuevo-volkswagen-passat> (Último acceso 15/09/2016).

- [11] IBÁÑEZ, *Los coches que se aparkan completamente solos*, 2013. URL: <http://www.motorpasionfuturo.com/ayudas-a-la-conduccion/los-coches-que-se-aparkan-completamente-solos> (Último acceso 15/09/2016).
- [12] MORRILLU, *¿Qué son las luces adaptativas y automáticas?*, 2012. URL: <http://www.circulaseguro.com/que-son-las-luces-adaptativas-y-automaticas/> (Último acceso 15/09/2016).
- [13] *Detector de ángulo muerto*. URL: <http://www.autopista.es/glosario/D/detector-de-angulo-muerto> (Último acceso 15/09/2016).
- [14] MORRILLU, *¿Qué son los sistemas de visión nocturna?*, 2013. URL: <http://www.circulaseguro.com/que-son-los-sistemas-de-vision-nocturna/> (Último acceso 15/09/2016).
- [15] ESTEBAN VISO, *¿Qué es la detección de las señales de tráfico?*, 2013. URL: <http://www.circulaseguro.com/que-es-la-deteccion-de-las-senales-de-traffic/> (Último acceso 15/09/2016).
- [16] IBÁÑEZ, *Sistemas de detección de la fatiga y falta de concentración al volante*, 2014. URL: <http://www.circulaseguro.com/sistemas-de-deteccion-de-la-fatiga-y-falta-de-concentracion-al-volante/> (Último acceso 15/09/2016).
- [17] LIC. MARGARITA RODRÍGUEZ SUÁREZ, *La rueda de las emociones de R. Plutchik*, 2012. URL: <http://soberanamente.com/la-rueda-de-las-emociones-de-r-plutchik/> (Último acceso 15/09/2016).
- [18] IVÁN PICO, *La rueda de las emociones, de Robert Plutchik*, 2016. URL: <http://psicopico.com/la-rueda-las-emociones-robert-plutchik/> (Último acceso 15/09/2016).
- [19] JOSÉ LUIS DÍAZ, ENRIQUE O. FLORES, *La estructura de la emoción humana: Un modelo cromático del sistema afectivo*, 2001. URL: <http://www.redalyc.org/pdf/582/58242403.pdf> (Último acceso 15/09/2016).
- [20] NORMA PATRICIA HERNÁNDEZ MACIP, *Aplicación de un Modelo Bidimensional de Estados Afectivos para la Comunicación entre Múltiples Usuarios*, 2006. URL: [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/hernandez\\_m\\_np/capitulo3.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/hernandez_m_np/capitulo3.pdf) (Último acceso 15/09/2016).
- [21] UNESPA, *Conducción Zen. Emociones, decisiones y conducción*, 2010. URL: [http://www.unespa.es/documentacion/videos/ESTUDIO\\_ZenDriving.pdf](http://www.unespa.es/documentacion/videos/ESTUDIO_ZenDriving.pdf) (Último acceso 15/09/2016).
- [22] DGT, *Instrucción 15/V-113 - Autorización de pruebas o ensayos de investigación realizados con vehículos de conducción automatizada en vías abiertas al tráfico en general*, 2016.
- [23] BOE, *Real Decreto 2822/1998, de 23 de diciembre, por el que se aprueba el Reglamento General de Vehículos*, 1999.
- [24] BOE, *Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal.*, 2000.

- [25] CATALDI, Z., LAGE, F., PESSACQ, R. Y GARCÍA MARTÍNEZ, R., *Ingeniería de software educativo*.
- [26] VANGOS PTERNEAS, *How To Use Kinect v2 Face Basics*, 2014How To Use Kinect v2 Face Basics. URL: <http://www.codeproject.com/Articles/855623/How-to-use-Kinect-v-Face-Basics> (Último acceso 15/09/2016)
- [27] VANGOS PTERNEAS, *Understanding Kinect Coordinate Mapping*, 2014. URL: <http://www.codeproject.com/Articles/769608/Understanding-Kinect-Coordinate-Mapping> (Último acceso 15/09/2016)
- [28] VANGOS PTERNEAS, *Kinect for Windows version 2: Color, depth and infrared streams*, 2014. URL: <http://www.codeproject.com/Articles/730842/Kinect-for-Windows-version-Color-depth-and-infra> (Último acceso 15/09/2016)
- [29] *Microsoft API*. URL: <https://msdn.microsoft.com/en-us/library/microsoft.kinect> (Último acceso 15/09/2016)
- [30] ERIK KLIMCZAK, *Blob Tracking: Kinect + OpenCV + WPF*, 2012. URL: <http://blogs.claritycon.com/blog/2012/11/blob-tracking-kinect-opencv-wpf/> (Último acceso 15/09/2016)
- [31] *Emgu CV Library Documentation*, 2016. URL: <http://www.emgu.com/wiki/files/3.1.0/document/html/8dee1f02-8c8a-4e37-87f4-05e10c39f27d.htm> (Último acceso 15/09/2016)
- [32] J.MACCORMICK, *How does Kinect work?*. Dickinson College.
- [33] DANIEL DE LA FLOR ACEITUNO, *Determinación del estado emocional de un conductor mediante el uso de imágenes infrarrojas*, 2015.
- [34] STISIM DRIVE, *STISIM Drive - Introduction*. URL: [http://web.mit.edu/16.400/www/auto\\_sim/Help/Introduction.htm](http://web.mit.edu/16.400/www/auto_sim/Help/Introduction.htm) (Último acceso: 16/06/2016).
- [35] Z. CATALDI, F. LAGE, R. PESSACQ Y R. GARCÍA MARTÍNEZ, *INGENIERIA DE SOFTWARE EDUCATIVO*. URL: <http://www.iidia.com.ar/rgm/comunicaciones/c-icie99-ingenieriasoftwareeducativo.pdf> (Último acceso 15/09/2016).
- [36] CARLOS BLE, *Diseño ágil con TDD*. URL: [http://librosweb.es/libro/tdd/capitulo\\_1/modelo\\_en\\_cascada.html](http://librosweb.es/libro/tdd/capitulo_1/modelo_en_cascada.html) (Último acceso 15/09/2016).